

Andreas Mutzke, Ralf Schneider, Gunnar Bandelow

**SDTrimSP-2D: Simulation of Particles Bombarding on a  
Two Dimensional Target  
Version 2.0**

**IPP 12/11**  
Nov, 2013

# SDTrimSP-2D: Simulation of Particles Bombarding on a Two Dimensional Target

## Version 2.0

A. Mutzke, R. Schneider, G. Bandelow

November 14, 2013

### Contents

<b>1. Introduction</b>	<b>3</b>
<b>2. Geometry</b>	<b>4</b>
<b>3. Change of cell volume</b>	<b>7</b>
<b>4. Target relaxation</b>	<b>10</b>
<b>5. Change of density and target fraction in a cell</b>	<b>14</b>
<b>6. Determination of the anisotropy weight factor</b>	<b>16</b>
<b>7. Comparison of results from SDtrimSP and SDTrimSP-2D</b>	<b>19</b>
<b>8. Two-dimensional examples with SDTrimSP-2D</b>	<b>20</b>
8.1. 5000 eV Xe on Si . . . . .	20
8.2. 6000 eV Ar on Si target with structure . . . . .	23
8.3. 6000 eV C on Si target with structure . . . . .	25
8.4. Influence of resolution . . . . .	27
8.5. 5000 eV W on C-target with incidence angle of $\alpha=30^\circ$ . . . . .	28
<b>A. Units of physics terms in the code</b>	<b>30</b>
<b>B. Search algorithm of local cell</b>	<b>31</b>
<b>C. Geometry of a binary collision</b>	<b>33</b>
C.1. Gaussian distribution of incidence angle . . . . .	34
C.2. Gaussian distribution of incidence energy . . . . .	34

*Contents*

<b>D. Global parameters</b>	<b>35</b>
<b>E. Input variables in 'tri.inp'</b>	<b>35</b>
E.1. Necessary input variables in 'tri.inp' . . . . .	35
E.2. Optional input variables in 'tri.inp' . . . . .	37
<b>F. Example of Inputfile 'tri.inp' and 'layer.inp'</b>	<b>43</b>

# 1. Introduction

The new program version is based on the one-dimensional TRIMM, TRIDYN and SDTrimSP (Static and Dynamic Trim for Sequential and Parallel computer) codes described in [1], [2], [3] and [4]. All these programs simulate the interaction of energetic particles with solids using the binary collision approximation.

The particle (projectile), impinging from outside on the solid (target), can be a neutral atom or an ion. If a projectile penetrates a solid target it will be scattered due to collisions with target atoms, which lead to an elastic energy loss and to a change of direction. In addition, the projectiles suffer an inelastic energy loss due to collisions with electrons. When the projectiles have lost all their energy, they are stuck in the target. Other possibilities are that the projectiles are backscattered after some collisions or that they are transmitted.

The energy lost by a projectile in a collision is transferred to a target atom (recoil), which itself can collide with other target atoms. If its energy is large enough the recoil can also leave the target (backward or transmission sputtering).

The movement of particles take place in three dimensions. The target is resolved in only one dimension in the program SDTrimSP. It consists of layers, which have different thickness and varying composition.

In this new 2-D version the target is resolved in two dimensions (SDTrimSP-2D), allowing to account for effects of roughness on the scattering and sputtering and to model dynamical morphological changes.

This report is an extension of the previous version (IPP 12/4).

## 2. Geometry

### 2. Geometry

The main difference between SDTrimSP [5] and SDTrimSP-2D is the mapping of the target geometry. In SDTrimSP the target is one dimensional (Fig. 1a) and treated as a stack of layers in X direction, while Y and Z direction are taken as infinite. The code can be used in two different modes: In 'static mode' the target is fixed, while in 'dynamical mode' the thickness of layers is changeable.

The expansion into a second dimension makes it possible to describe target surface profiles and their dynamic change. The target in SDTrimSP-2D is therefor a regular grid in X- and Z-direction, while Y direction stays infinite, shown in Fig. 1b.

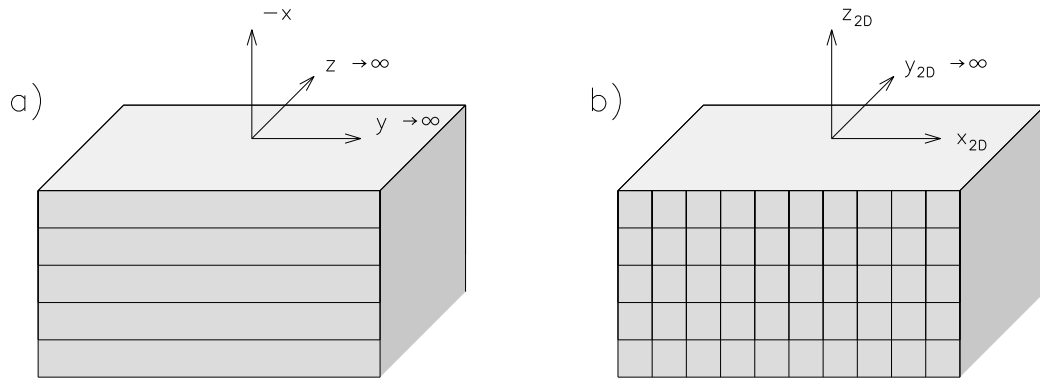


Figure 1: Geometry of one dimensional target a) and two dimensional target b)

A right-handed coordinate system  $(X_{2D}, Y_{2D}, Z_{2D})$  is used in the new code to describe particle positions and cell geometry. Fig. 2 shows the definition of polar and azimuthal angles (blue) in comparison to the coordinate system used in SDTrimSP (red) and SDTrimSP-2D (black). The coordinate origin is fixed and not movable, like in SDTrimSP.

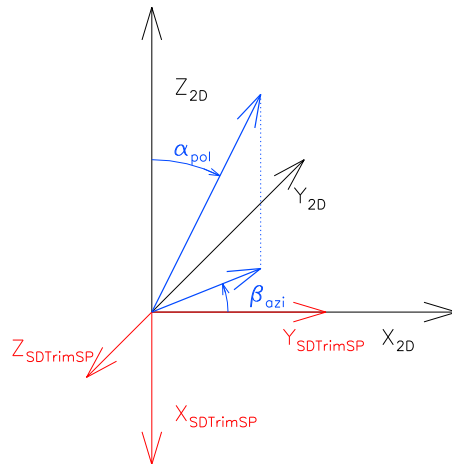


Figure 2: Definition of the coordinates, polar ( $\alpha$ ) and azimuthal ( $\beta$ ) angle

Each cell in the two dimensional grid has 4 active surfaces and directions (left, right, bottom and top), that are described in Fig. 3. The surface-indices are counted from one to four in the program, while for better handling direction-indices are ranging from -2 to +2.

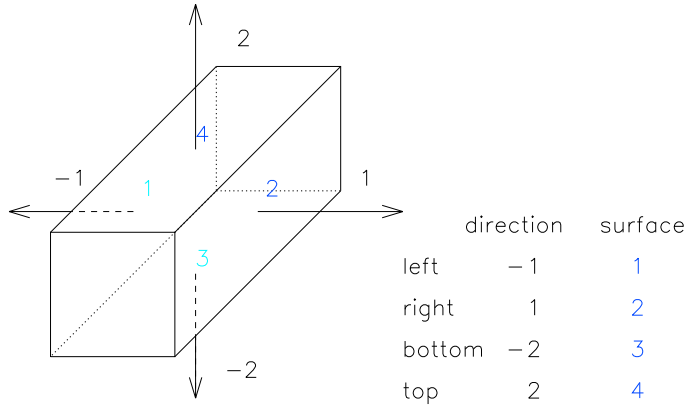


Figure 3: Indices of directions and surfaces of one cell used in the program.

To be able to treat the 2D problem additional features were introduced. There are two kinds of cells, depending on their location in the target labeled as active or passive. Passive cells, P, can not change their geometry and their volume is constant. If these cells swell or shrink, a relaxation is only possible due to volume- and particle-fluxes to neighboring cells, Fig. 4. The handling of this process is described in section 4.

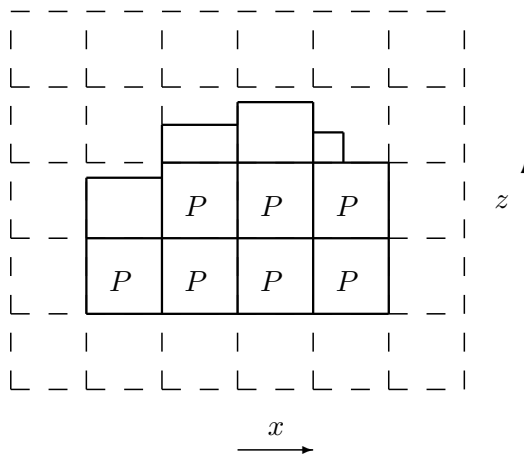


Figure 4: Target geometry with passive (P) or active grid cells.

## 2. Geometry

Active cells, A, have at minimum one open boundary and their size can even be smaller than the mesh, thus their volume isn't constant in contradiction to passive cells. Active cells can only have one or two open boundaries. Some possible settings of active cells are shown in Fig. 5. Case (a) demonstrates an active surface cell with open boundaries to the upper and right side. Due to the neighboring active cell on the left side this boundary is not open. In case (b) the active cell is pinched between two passive cells. The resulting open boundaries are at top and bottom. Case (c) and (d) are examples for active cells with only one open boundary. This is possible, if the active cell has three or only one neighboring cell.

Growth and shrinkage processes of the target can only occur at open boundaries.

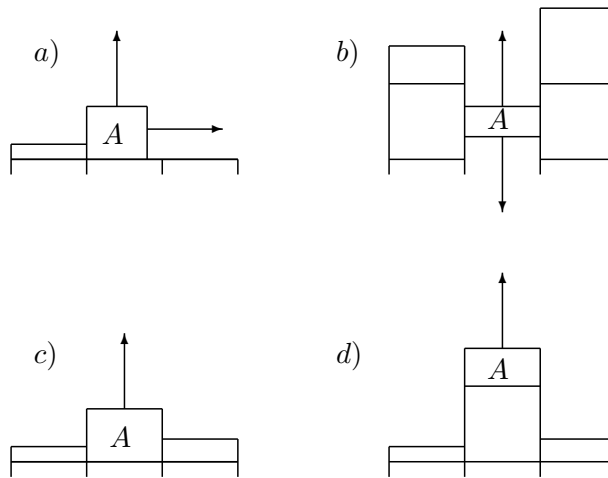


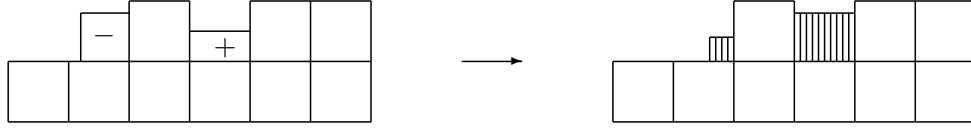
Figure 5: Examples of active cells (A) with two (a, b) and one (c, d) open boundaries.

During a simulation run cells might change their status from active to passive and backwards, depending on the movement of particles in the surrounding cells. Three different examples of active cell behaviour are shown in Fig. 6.

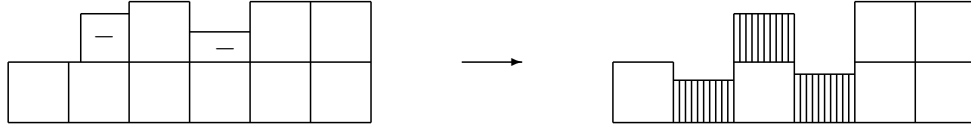
Case 1 is the simple volume change of active cells, that are able to swell (+) or shrink (-) inside the grid. If the reduction of the volume is greater than the cell-volume the cell is deleted and the next cell gets the status active (boundary) cell (case 2). If the new volume is greater than the standard cell-volume new active cells are created (case 3). This calculation is performed in subroutine *'change\_vol'*.

A closer look into calculation of the cell volume change is given in the next chapter.

case 1: cells swell or shrink



case 2: cells shrink until deletion and neighboring cells shrink further



case 3: cells swell to maximum and new cells are created

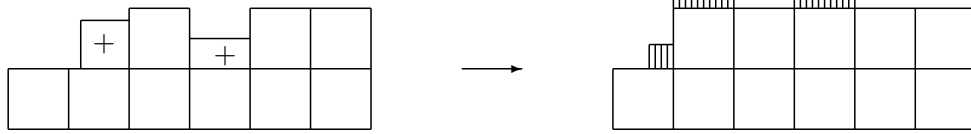


Figure 6: Change of geometry of active cells, right side: cells after relaxation.

### 3. Change of cell volume

Due to the constant flux of particles onto the target during a simulation, its composition and thus the volume of layers (cells) changes dynamically.

The number of implanted projectiles  $u_{im}$ , the number of recoils transferred into  $u_{in}$  and the number of atoms removed from a layer (cell) into another layer (cell) or sputtered  $u_{out}$  are determined after the bombardment of a target with  $n_r$  pseudo-particles (in program: `nr_pproj`). The probability that one incident particle changes the number of atoms in one layer or cell is  $N_0$ :

$$N_0(j) = \frac{u_{im}(j) + u_{in}(j) - u_{out}(j)}{n_r} \quad j \dots \text{species of atoms } (j = 1 \dots ncp) \quad (3.1)$$

The composition of the target is updated after one fluence step, which is the quotient of fluence  $flc$  and number of fluence steps  $maxhist$ :

$$\Delta flc = \frac{flc}{maxhist} \quad (\text{in program: } fluc\_step)$$



### 3. Change of cell volume

The atomic composition of target determines the atomic fraction  $qu$  and the number of species is  $ncp$ . The sum of all atomic fractions  $qu$  is one.

$$\sum_{j=1}^{ncp} qu(j) = 1$$

In some cases it is necessary to define a maximum allowed atomic fraction. This maximum allowed atomic fraction  $qu_a^{max}$  can be defined in order to simulate local saturation phenomena, see [2].

If a maximum for one species (a) is given, the number of particles can be calculated accordingly:

$$N_m(a)^{max} = \frac{qu_a^{max}}{1 - qu_a^{max}} \cdot \left( \sum_{j=1}^{ncp} N_0(j) - N_0(a) \right)$$

$$N'_0(a) = \max(N_m(a)^{max}, N_0(a))$$

If two maxima  $qu_a^{max}$  and  $qu_b^{max}$  are given for two species a and b, then the number of particles are :

$$N'_0(a) = \frac{Q_a \cdot Q_b + Q_a}{1 - Q_a \cdot Q_b} \cdot \left( \sum_{j=1}^{ncp} N_0(j) - N_0(a) - N_0(b) \right)$$

$$N'_0(b) = Q_b \cdot \left( \sum_{j=1}^{ncp} N_0(j) - N_0(a) - N_0(b) + N'_0(a) \right)$$

with:

$$Q_a = \frac{qu_a^{max}}{1 - qu_a^{max}}$$

$$Q_b = \frac{qu_b^{max}}{1 - qu_b^{max}}$$

In the following 1D and 2D solution to the change of 'volume' are described.

## SDTrimSP

In the program SDTrimSP [5] the new absolute number of atoms after one fluence-step  $\Delta flc$  is  $N_1^{1D}(j)$  (normalized to  $\Delta x \cdot \Delta y$ ), i.e. it is equivalent to the areal density. The thickness of one layer is  $\Delta z$ , the atomic fraction is  $qu$  and the number density of the pure solid is  $\rho_0$ . The new thickness of the layer  $z_{new}$  and the new volume  $V_{new}$  of one layer can be calculated according to the following formula:

$$V_{new} = z_{new} \cdot \Delta x \cdot \Delta y = \sum^j \frac{N_1^{1D}(j)}{\rho_0(j)} \cdot \Delta x \cdot \Delta y$$

$$z_{new} = \sum^j \frac{N_1^{1D}(j)}{\rho_0(j)}$$

with:

$$N_1^{1D}(j) = \Delta N + N = N_0(j) \cdot \Delta flc + qu_{old}(j) \cdot \rho_{old} \cdot \Delta z$$

## SDTrimSP-2D

In the program with 2D-targets the number of atoms in one cell is the sum of implanted projectiles, the number of recoils transferred into, the number of atoms removed from the cell (eq. 3.1) and the number of existing atoms (old) after one fluence step is:

$$N_1(j) = N_0(j) \cdot \Delta flc \cdot A_{flc} + qu_{old}(j) \cdot \rho_{old} \cdot V_c$$

The new volume is:

$$V_{new} = \Delta V_c + V_c = \sum^j \frac{N_1(j)}{\rho_0(j)}$$

If  $V_c$  and  $V_{new}$  are known, the change of the volume  $\Delta V_c$  can be calculated.

To calculate the volume change a value for the real beam surface is needed in simulations with SDTrimSP-2D. Besides one has to keep in mind that the target in SDTrimSP-2D can be one (like SDtrimSP) or two dimensional.

For 1-D cases, see Fig. 18, the cells are so large that they can be treated as a layer. In this case the  $\Delta x$  of the cell must be much greater than the maximum of x-values during a particle cascade. This is the same geometry as in the code SDTrimSP. The infinite extension in the y-direction is achieved by a large value of  $\Delta y_{cell}$  of all cells. In this case, the real beam surface  $A_{flc}$  can be calculated using the cell geometry:

$$A_{flc} = \Delta x_{cell} \cdot \Delta y_{cell} \quad (1-D)$$

In the 2-D case, the real beam surface  $A_{flc}$  can be calculated using the cell and beam geometry :

$$A_{flc} = \Delta x_{beam} \cdot \Delta y_{cell} \quad (2-D) \quad (\text{in program: } A_{flc} \dots \text{beam\_geo\_fac})$$

#### 4. Target relaxation

### 4. Target relaxation

Implantation of projectiles and relocation or sputtering of recoils caused by bombardment produce vacancies and additional atoms in the target. Therefore the cell volume is changed and the target is allowed to relax.

#### SDTrimSP

In the 1D model, the layers can swell and shrink. A new target thickness  $z_{new}$  can be calculated. The starting point of the x-axis is removed on the surface. Fig. 7 shows the simple relaxation of a 1D-target with three layers ( $d_1, d_2, d_3$ ), which swell (+) or shrink (-). The thickness of the whole target is the sum of all layer-thickness. The origin of the coordinate system is moved to the new surface.

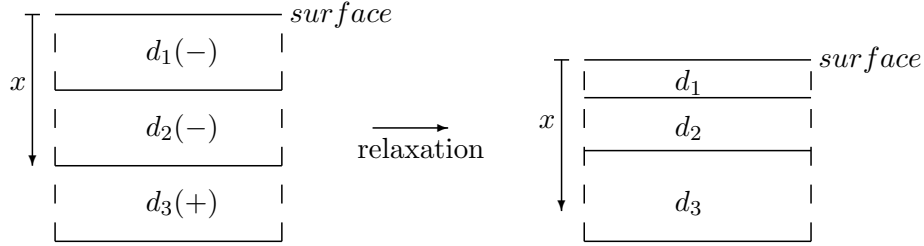


Figure 7: Relaxation of 1D-target with three layers ( $d_1, d_2, d_3$ )

#### SDTrimSP-2D

The relaxation method used in 1D is not suitable for the 2D case. Therefore, a new concept of relaxation had to be found:

The dynamic relaxation of the target is acting on the volumes. Particles are transported into cells in the neighborhood. The assumption is that the cells are squares and the increase or decrease of their volumes are the sum of the transfers from all directions. the change of volume  $\Delta V_c$  is given and can be splitted in  $V_l, V_r, V_t$  and  $V_b$ . An example of a volume change is given in Fig. 8 with the expansion of a cell.

The additional volume can be calculated accordingly:

$$\begin{aligned} \Delta V_c &= V_l + V_r + V_t + V_b && (r\dots right, l\dots left, t\dots top, b\dots bottom) \\ \Delta V_c &= (dz \cdot dl + dz \cdot dl + dx \cdot dl + dx \cdot dl) \cdot dy \\ \Delta V_c &= dl \cdot (2 \cdot dz + 2 \cdot dx) \cdot dy \end{aligned}$$

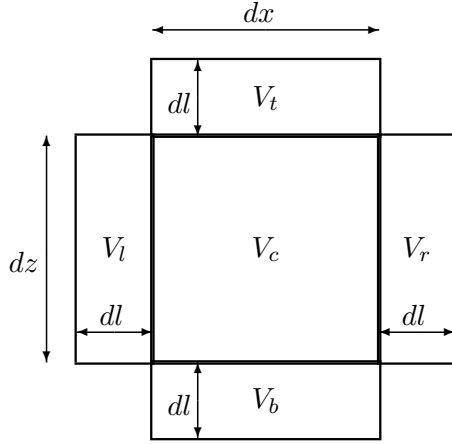


Figure 8: The expansion of the volume

Under the assumption of an equally expanding cell towards the four surfaces:

$$\begin{aligned} V_{l/r} &= dl \cdot dz \cdot dy \\ V_{t/b} &= dl \cdot dx \cdot dy, \end{aligned} \quad (4.2)$$

it is possible to eliminate  $dl$  and calculate the volume changes:

$$\begin{aligned} V_{l/r} &= \Delta V_c \cdot \frac{dz}{2(dz + dx)} \\ V_{t/b} &= \Delta V_c \cdot \frac{dx}{2(dz + dx)}. \end{aligned} \quad (4.3)$$

To get a consistent treatment of the target, volume changes have to be converted into 'fluxes'. At the beginning of a fluence step the volume of every cell is smaller/greater (active) or equal (passive cell) to the grid cell size and therefore the divergence of the fluxes between cells is zero.

$$\nabla \cdot \vec{F} = 0$$

After bombardement of the target, there might exist defects (sinks) or particles were moved, respectively implanted (sources). Thus, the target is no longer divergence-free. The additional flux is proportional to the volume change.

$$\nabla \cdot \vec{F} = Q = k \cdot \Delta V \quad \text{k...dimension-factor}$$

The relaxation process now has the task to eliminate sources and sinks to restore a divergence-free target.

$$\Delta V(\tau \rightarrow \infty) = 0$$

#### 4. Target relaxation

The elimination of sources and sinks and therefore the calculation of fluxes is carried out by a diffusion process with a pseudo relaxation-time  $\tau'$ . The volume fluxes are calculated in the subroutine 'flux\_vol'.

$$\frac{\partial V}{\partial \tau'} = \frac{\partial D \cdot \frac{\partial V}{\partial x}}{\partial x} + \frac{\partial D \cdot \frac{\partial V}{\partial y}}{\partial y} + \frac{\partial D \cdot \frac{\partial V}{\partial z}}{\partial z}$$

If the target is two dimensional (like in SDTrimSP-2D) the equation has only two terms:

$$\frac{\partial V}{\partial \tau'} = \frac{\partial D \cdot \frac{\partial V}{\partial x}}{\partial x} + \frac{\partial D \cdot \frac{\partial V}{\partial z}}{\partial z} \quad (4.4)$$

The finite-difference form is upstream in pseudo-time and centered in space:

$$\begin{aligned} \frac{\Delta V}{\Delta \tau'} &= \frac{D_{i+1/2} \cdot (V_{i+1} - V_i) - D_{i-1/2} \cdot (V_i - V_{i-1})}{\Delta x^2} \\ &+ \frac{D_{k+1/2} \cdot (V_{k+1} - V_k) - D_{k-1/2} \cdot (V_k - V_{k-1})}{\Delta z^2} \end{aligned}$$

with:

$$\begin{aligned} \Delta V(\tau' = 0) &= \Delta V_c \quad (\text{initial condition, after bombardment}) \\ \Delta V(\tau' \rightarrow \infty) &= 0 \quad (\text{end condition, after relaxation}) \end{aligned}$$

If  $\Delta x = \Delta z$  and  $\tau = \tau'/\Delta x^2$  then:

$$\begin{aligned} \frac{\Delta V}{\Delta \tau} &= D_{i+1/2} \cdot V_{i+1} - D_{i+1/2} \cdot V_i - D_{i-1/2} \cdot V_i + D_{i-1/2} \cdot V_{i-1} \\ &+ D_{k+1/2} \cdot V_{k+1} - D_{k+1/2} \cdot V_k - D_{k-1/2} \cdot V_k + D_{k-1/2} \cdot V_{k-1} \end{aligned}$$

The reduction of tension (or  $\Delta V_c$ ) of the target takes place mainly in the direction of the surface. It is possible to control this anisotropy with different diffusion-coefficients or the weight-factors *wig* (see chapter. 6). Replacing the Volume  $V$  by  $V_l, V_r, V_t, V_b$  and the diffusion-coefficient  $D$  by  $wig_l, wig_r, wig_t, wig_b$  one gets:

$$\begin{aligned} \frac{\Delta V}{\Delta \tau} &= wig_{l,i+1} \cdot V_{l,i+1} - wig_{r,i} \cdot V_{r,i} - wig_{l,i} \cdot V_{l,i} + wig_{r,i-1} \cdot V_{r,i-1} \\ &+ wig_{b,k+1} \cdot V_{b,k+1} - wig_{t,k} \cdot V_{t,k} - wig_{b,k} \cdot V_{b,k} + wig_{t,k-1} \cdot V_{t,k-1} \end{aligned}$$

All terms of the finite-difference equation can be written in a flux-form:

$$\frac{V_{\tau+1,i} - V_{\tau,i}}{\Delta\tau} = F_{l,i+1} - F_{r,i} - F_{l,i} + F_{r,i-1} + F_{b,k+1} - F_{t,k} - F_{b,k} + F_{t,k-1}$$

Here, F...volume-flux, r...right, l...left, t...top, b...bottom.

Finite-difference equation for a cell with number  $C_{Nr}=0$  according to Fig. 9 is:

$$V_{0,\tau+1} = V_{0,\tau} + (F_{2,l} - F_{0,r} + F_{1,r} - F_{0,l} + F_{4,b} - F_{0,t} + F_{3,t} - F_{0,b}) \cdot \Delta\tau \quad (4.5)$$

With help of equations 4.3 the fluxes are:

$$\begin{aligned} F_{0,l} &= \Delta V_{0,\tau} \cdot \frac{dz_0}{2(dz_0 + dx_0)} \cdot wig_{l,0} \\ F_{0,r} &= \Delta V_{0,\tau} \cdot \frac{dz_0}{2(dz_0 + dx_0)} \cdot wig_{r,0} \\ F_{0,t} &= \Delta V_{0,\tau} \cdot \frac{dx_0}{2(dz_0 + dx_0)} \cdot wig_{t,0} \\ F_{0,b} &= \Delta V_{0,\tau} \cdot \frac{dx_0}{2(dz_0 + dx_0)} \cdot wig_{b,0} \end{aligned}$$

If the target is periodical in x-direction cells on the left side of the target are directly connected to their counterparts on the right side.

The solution of equations 4.4 and 4.5 are the transport fluxes (F) represented as volumes. If the solution is stationary, a full relaxation of the target is achieved. This is calculated in the subroutine '*flux\_vol*'.

$$\Delta V(\tau \rightarrow \infty) = 0$$

$$F_{right} = \sum_{\tau=0}^{\infty} (F_{2,l} - F_{0,r}) \quad (4.6)$$

$$F_{left} = \sum_{\tau=0}^{\infty} (F_{1,r} - F_{0,l}) \quad (4.7)$$

$$F_{top} = \sum_{\tau=0}^{\infty} (F_{4,b} - F_{0,t}) \quad (4.8)$$

$$F_{bottom} = \sum_{\tau=0}^{\infty} (F_{3,t} - F_{0,b}) \quad (4.9)$$

According to volume-flux in x and z directions the particle transport is calculated to the neighbour cells.

## 5. Change of density and target fraction in a cell

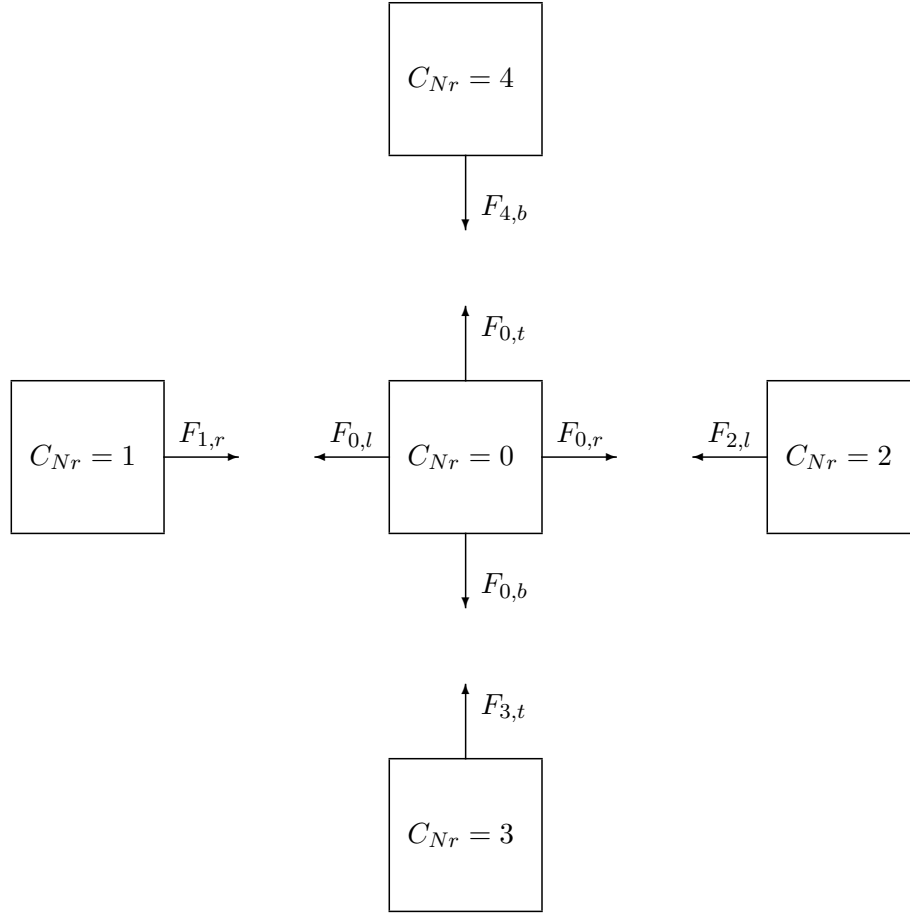


Figure 9: Volume-flux from and to a cell with cell-number  $C_{Nr}=0$

## 5. Change of density and target fraction in a cell

In almost all simulations exist more than one specie. Thus, density and atomic fraction in the target and the single cells have to be taken into account. Especially the relaxation process has to consider the physical limits.

The number of particles, which originate from the neighboring cells ( $m$ ) is

$$N_{in}(j) = \sum_m \frac{F_{in}(m)}{V(m) + dV_c(m)} \cdot N_2(m, j) \quad j \dots \text{species of atoms},$$

while the incoming volume is

$$dV_{in} = \sum_m F_{in}(m).$$

The Volume of a passive cell in the target is constant and independent of flux. The number of particles in the cell is:

$$N_3(j) = N_{in}(j) + \frac{(V_c - dV_{in})}{V_c + dV_c} \cdot N_2(j).$$

The number of particles in the active cell (surface-cell) and its additional volume is:

$$N_3(j) = N_{in}(j) + \frac{(V_c - dV_{in})}{V_c + dV_c} \cdot N_2(j) + \frac{dV_{add}}{V_c + dV_c} \cdot N_2(j)$$

$$dV_{add} = \sum_{ob} F_{out}(ob) \quad ob...flux\ over\ open\ boundary$$

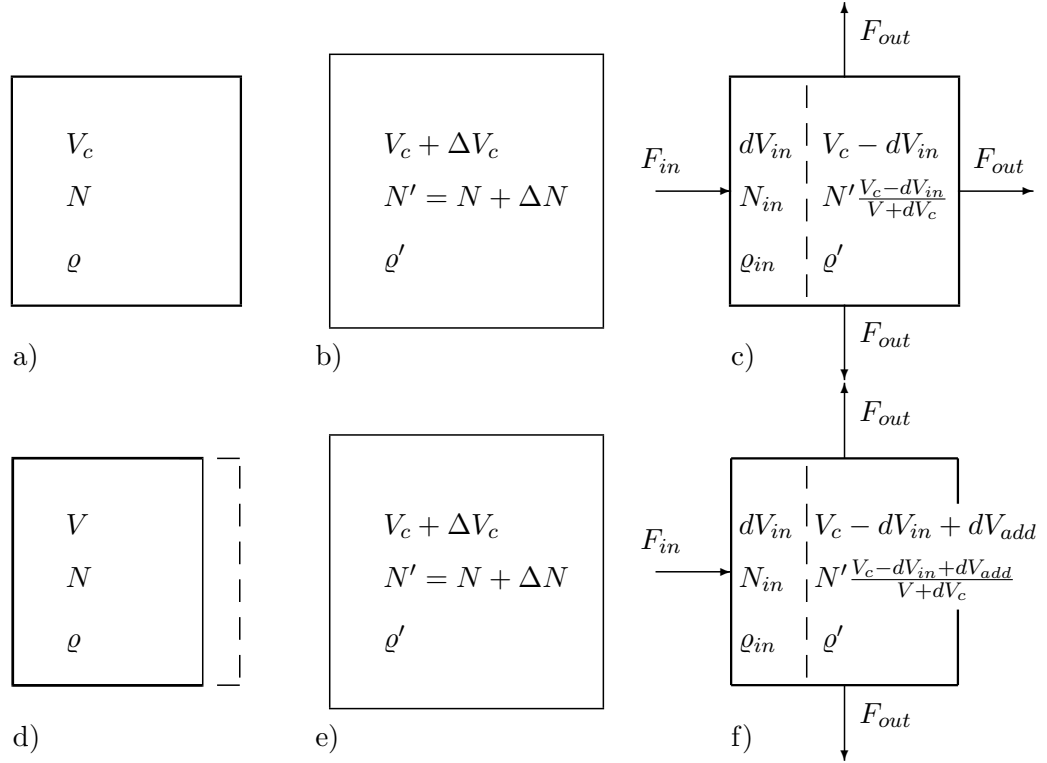


Figure 10: Parameters of a passive cell a) before bombardment, b) after collision and c) after relaxation,  $V(a) = V(c)$  and of an active cell with right open boundary d) before bombardment e) after collision and f) after relaxation,  $V(d) < V(f)$ .



## 6. Determination of the anisotropy weight factor

The new mean atomic density in each cell is therefore:

$$\frac{1}{\rho_{new}} = \sum_{k=1}^n \frac{qu(k)}{\rho_0(k)}$$

with composition

$$qu(j) = N_3(j) / \sum_{k=1}^{ncp} (N_3(k)) \quad j \dots \text{species of atoms} \quad k \dots \text{number of all species.}$$

## 6. Determination of the anisotropy weight factor

The relaxation is strongly depending on the used weighting scheme, which will be described in this section.

First step is a categorization of all cells in terms of their distance to the surface. Different levels are introduced, where the surface cells get the value one. The next 'layer' of cells connected to those surface cells, get the value two and so on. A smaller level corresponds to a cell closer to the surface. Figure 11 shows the procedure for an example target.

				1	1				
		1	1	2	2	1			
1	1	2	2	3	3	2	1	1	1
			$i_1$	$i_0$	$i_2$				
1	2	3	3	4	4	3	2	2	1
			$i_3$						
1	2	3	4	5	5	4	3	2	1

Figure 11: Determine levels for cells in the target.

Once the level of all cells are known, different rules for weighting factors can be applied. The weighting of the flux-connections,  $wig$ , between neighbour cells is depending on the level of the cells (see Fig. 11 and Fig. 12). If the level is smaller than the level of the neighbour cell then the weighting is 0.5, e.g. between cell  $i_0$  and  $i_3$ . Are the levels equal then the weighting is 1, e.g. between cell  $i_0$  and  $i_2$ . Is it greater then the value is 1. as well.

Only the surface cells get a special weighting. The weighting is 0.25 for the open boundary (no neighbour cell) and the weighting for the opposite side of the open boundary is 0.375. This choice produces more stable numerical solutions.

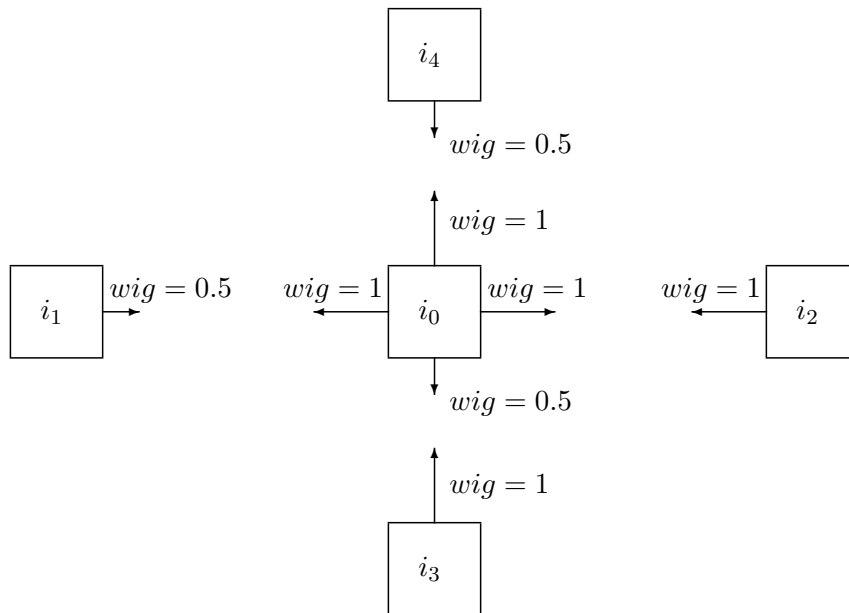


Figure 12: Determine the anisotropy weight factors  $wig$  for cell  $i_0$

6. Determination of the anisotropy weight factor

The difference between the solutions with an isotropic and anisotropic weighting scheme is small for one expanded cell and for one fluence step, but important for the whole target. This is caused due to non-local effects of the relaxation process across neighbours. In contradiction to the Eckstein ansatz in 1D the swelling happens not only perpendicular to the surface, but also to the sides ('smearing').

To show the influence of the weighting scheme on the simulation a silicon target was prepared (see Fig. 13) and bombarded with 5 keV Argon atoms under normal incidence.

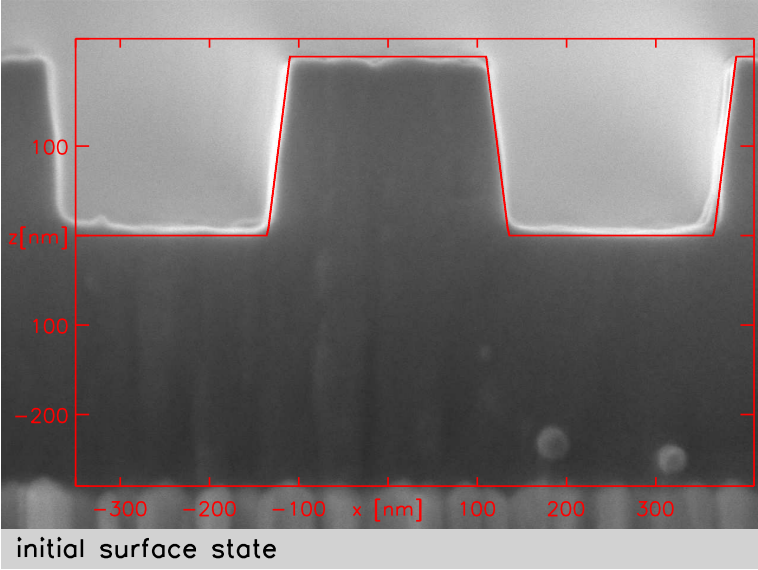


Figure 13: SEM images of a cross-section of initial state and model surface (red)

Fig. 14 shows simulation of this surface without (black) and with weighting (red line). A good agreement is achieved only with the use of anisotropic relaxation.

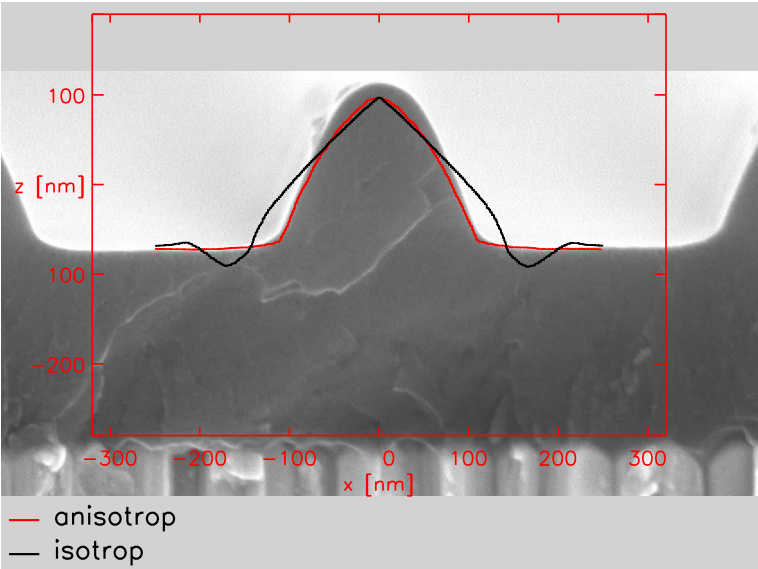


Figure 14: Comparison of calculated surface without (black) and with weighting (red) with experiment [6] 6 keV Ar on Si normal incident

## 7. Comparison of results from SDtrimSP and SDTrimSP-2D

In case of a static calculation (no change of target) a two dimensional target and its resolution are irrelevant, unless there are existing profiles in x and z directions. Due to possibility of running SDTrimSP-2D in a one dimensional mode direct comparisons with SDTrimSP were performed. A first dynamical test is to simulate a 1-D target, which is shown in Fig. 15.

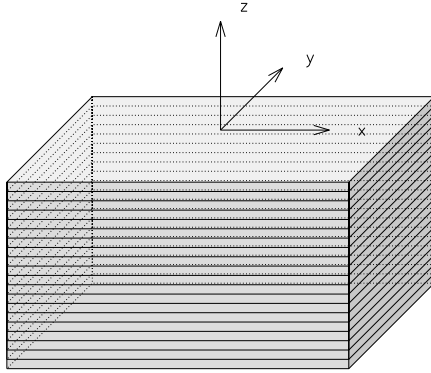


Figure 15: Geometry of one dimensional target in the 2D-program, according to SDTrimSP

The target consist of  $Ta_2O_5$  and was bombarded with  $1000\text{ eV He}$  without implantation of He. Fig. 16 shows the surface elevations, the depth profiles, the sputtering yields and the atomic fraction at surface for SDTrimSP and SDTrimSP-2D. The differences between the results are very small.

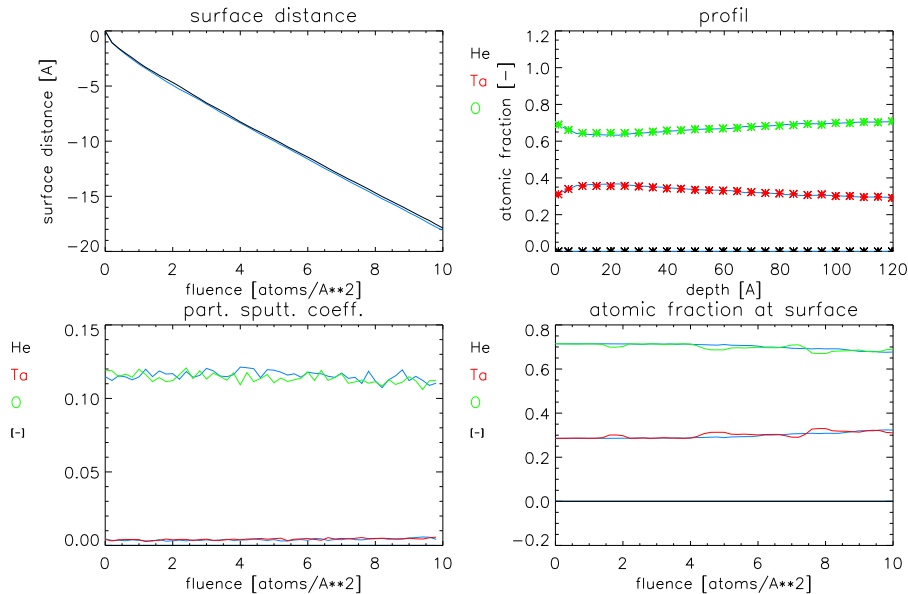


Figure 16:  $1000\text{ eV He}$  on  $Ta_2O_5$  , Comparison of results from SDTrimSP and from SDTrimSP-2D (blue)

## 8. Two-dimensional examples with SDTrimSP-2D

Another dynamical testcase is the bombardment of silicon target with 2000 eV Xe. In this simulation, Xe is implanted and the damage-driven-diffusion is used to simulate the outgasing process of Xe. The comparison of the results from simulations with SDTrimSP and SDTrimSP-2D shows Fig. 17. Again, the agreement is very good.

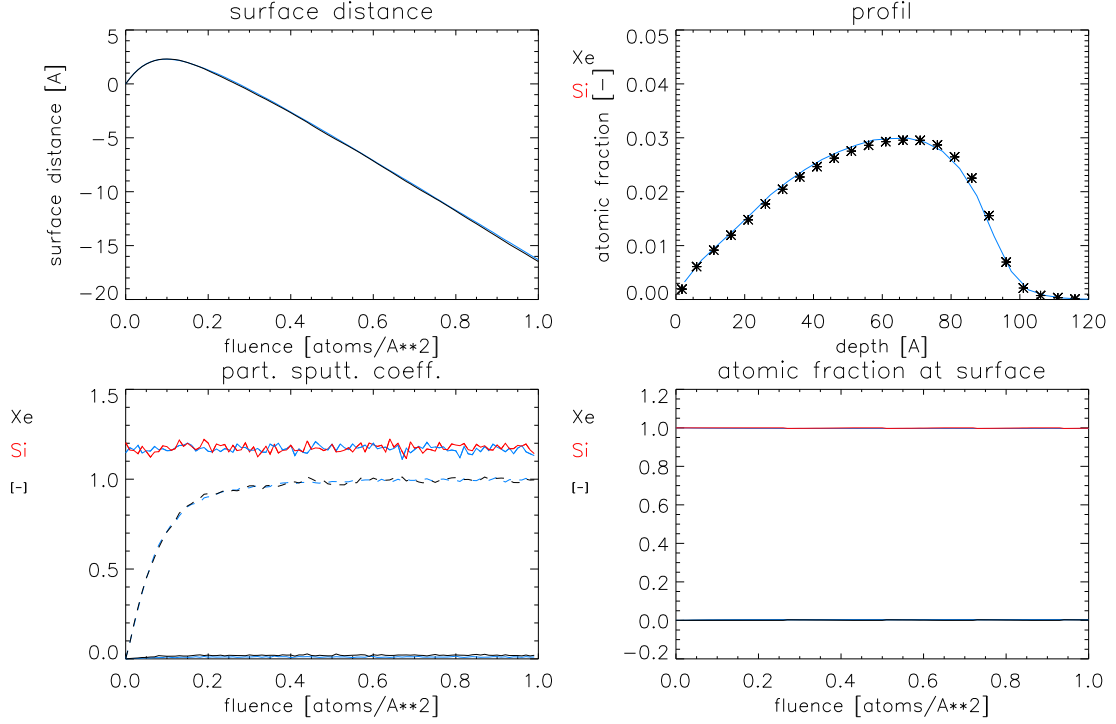


Figure 17: 2000 eV Xe on Si, Comparison of results from SDTrimSP-2D and from SDTrimSP (blue)

## 8. Two-dimensional examples with SDTrimSP-2D

The new capabilities of SDTrimSP-2D are shown in this section for different examples.

### 8.1. 5000 eV Xe on Si

One example is the bombarding of a silicon-target with xenon. The thickness of cells is  $\Delta z = 10 \text{ \AA} = 1 \text{ nm}$ . The energy of the beam is 5000 eV, the fluence is  $2.0 \text{ atoms/\AA}^2$  and the cutoff-energy is 1.0 eV.

The decrease of surface (sputter depth) in the model SDTrimSP is:

$$SD_{1D} = 60 \text{ \AA} \quad (8.10)$$

The 2-D-Model in case of 1-D-simulation ( $\Delta x, \Delta y \gg \Delta z$ ), see Fig. 15 and Fig. 18, we get a decrease of the surface:

$$SD_{2D} = 60 \text{ \AA} \quad (8.11)$$

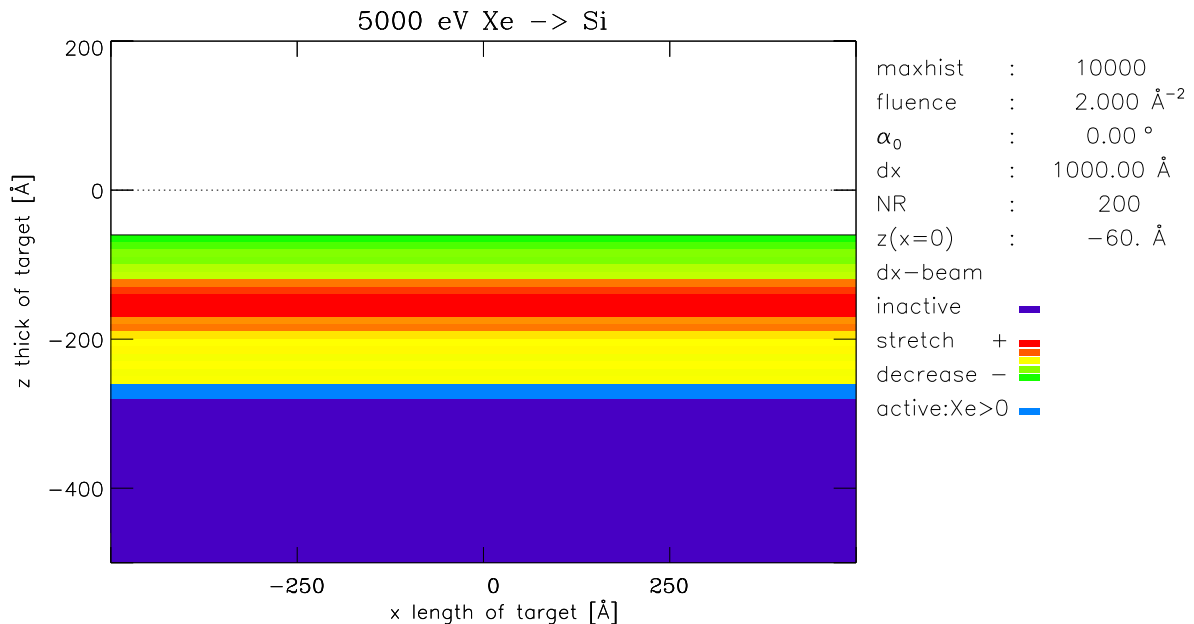


Figure 18: The target simulation with SDTrimSP-2D in case of 1-D-structure

The change of target in case of a 2-D simulation ( $\Delta y \gg \Delta x, \Delta z$ ) with different beam widths are shown in Fig. 19.

The decrease of surface or maximum sputtered depth (SD) in the center of the beam  $x = 0$ ,  $y = 0$  are listed for various beam widths in Tab. 1. In the case of very small width of the beam the calculated depth of material loss is greater, than in the cases with broader beams. The simulation with a periodical beam (indefinitely) produce the same results as the 1-D calculation.

nr.	$\Delta x$ of beam	maximum sputtered depth (SD)
1	50 Å	76 Å
2	100 Å	79 Å
3	200 Å	72 Å
4	400 Å	66 Å
5	indefinitely	60 Å
6	1-D-layers	60 Å

Table 1: Maximum of sputtered depth in the center of beam

8. Two-dimensional examples with SDTrimSP-2D

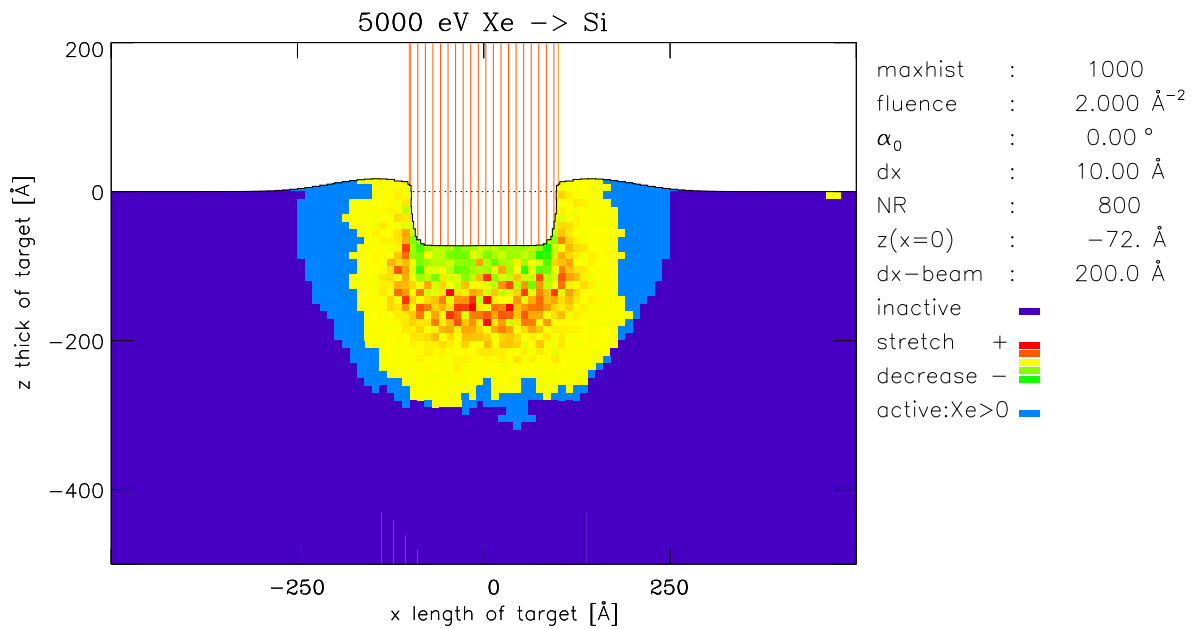
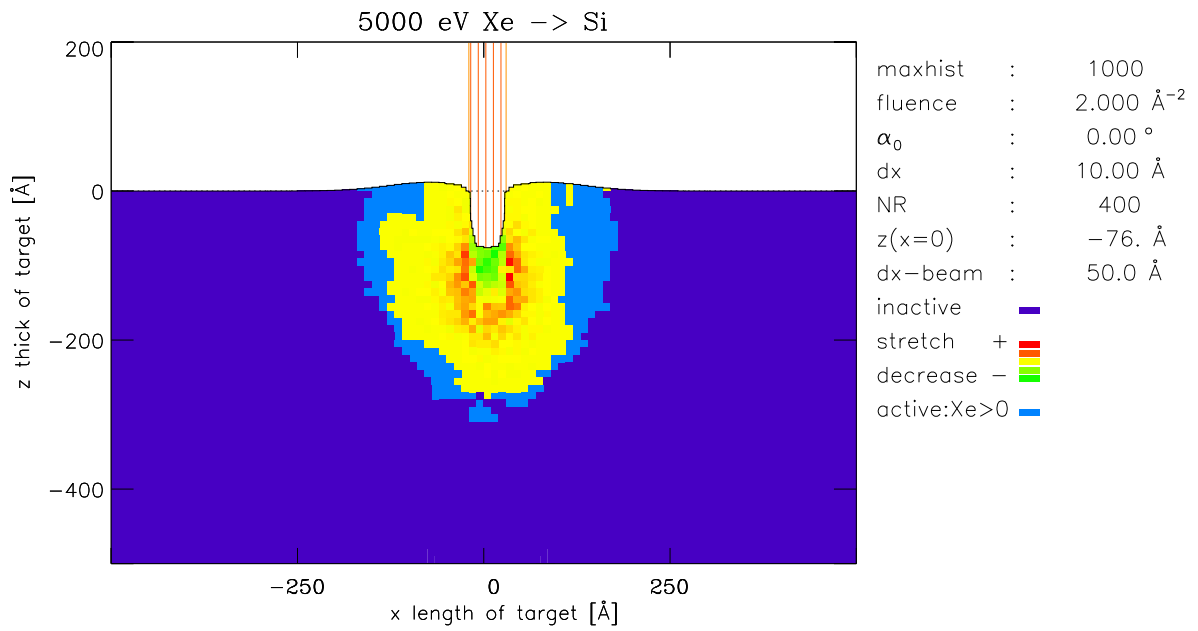


Figure 19: The target simulation  $\Delta x_{beam} = 50 \text{ \AA}$  and  $\Delta x_{beam} = 200 \text{ \AA}$

## 8.2. 6000 eV Ar on Si target with structure

The structure of the used target (pitch grating) is shown in Fig. 20.

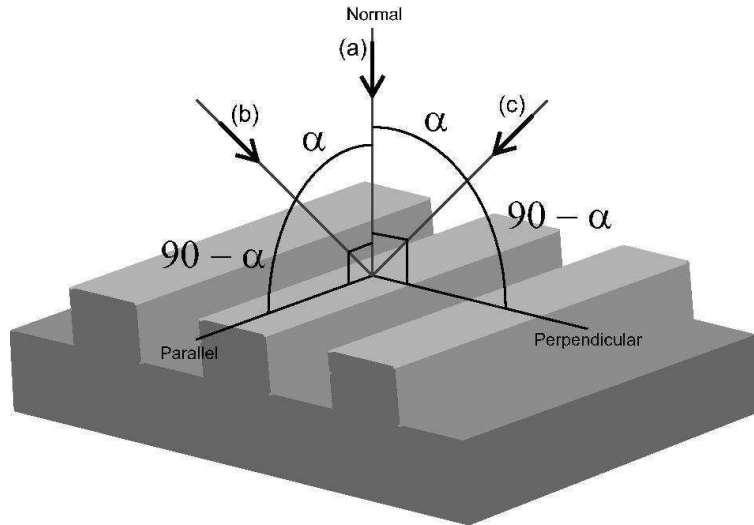


Figure 20: Structure of pitch grating and direction of normal incident (a)

Fig. 13 shows the cross-section at the beginning of bombardments. The nano-structured specimen is fabricated on a Si wafer with an intermediate Ta layer with thickness of 650 nm. The Ta layer is used as a reference marker to allow quantitative measurements of the Si layer thick. With increasing fluence the target shrinks. The agreement of calculated surface (red line) after bombardment with the experiment is very good, Fig. 21 and 22 a-b.

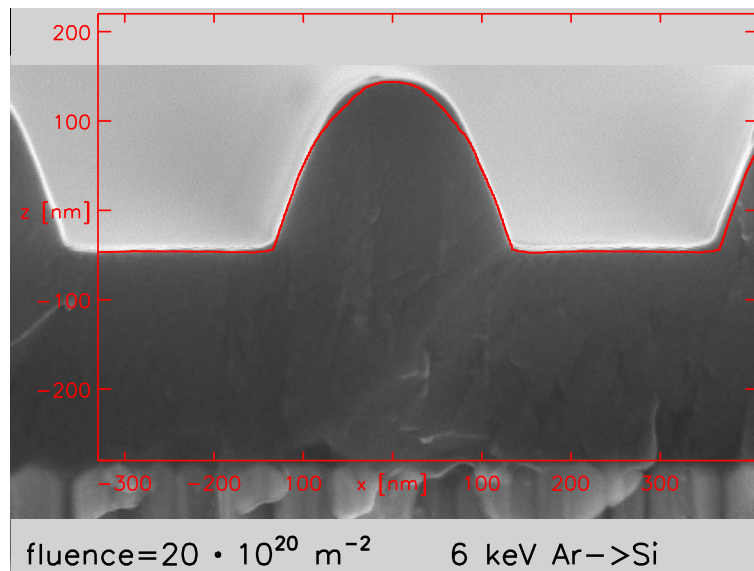


Figure 21: Comparison of calculated surface (red) with experiment from [6] of 6 keV Ar on Si normal incident at fluence  $20 \cdot 10^{20} \text{ atoms/m}^2$



## 8. Two-dimensional examples with SDTrimSP-2D

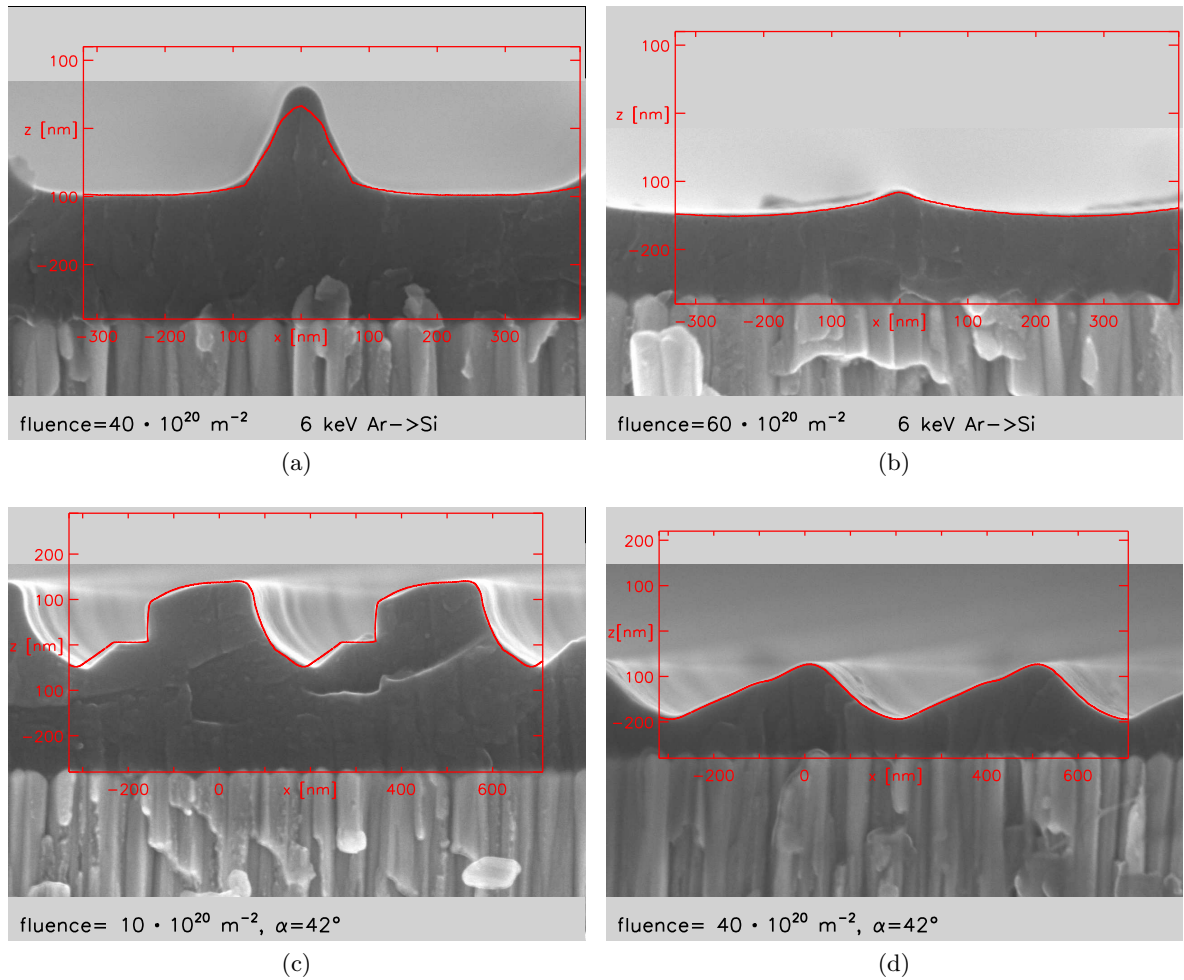


Figure 22: Comparison of calculated surface (red) with experiment from [6] of 6 keV Ar on Si.

The next experiment is the bombardment of the same target at an incident angle  $\alpha = 42^\circ$ , Fig. 22 c and 22 d. The agreement of calculated surface (red line) with the experiment is also very good.

### 8.3. 6000 eV C on Si target with structure

The structure of the used target (pitch grating) is the same as in Section 8.2 and in Fig. 20. If a silicon-target is bombarded with carbon, material is sputtered and removed. Fig. 23 shows the comparison of simulated surface with the experiment for a fluence of  $85 \cdot 10^{20} \text{ atoms/m}^2$ . The agreement of the calculated surface (red line) with the experiment is very good.

The comparisons of experimental data and simulation of areal density are shown in Fig. 24. The areal density of C atoms is growing due to the implantation. The areal density of Si is decreased due to to sputtering.

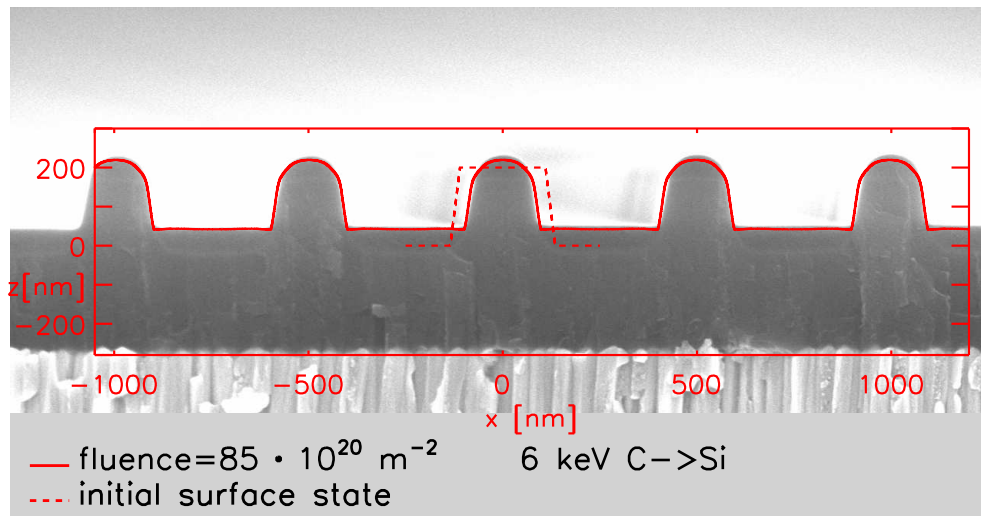


Figure 23: Comparison of calculated surface (red) with experiment [8] of 6 keV C on Si normal incident at fluence  $85 \cdot 10^{20} \text{ atoms/m}^2$

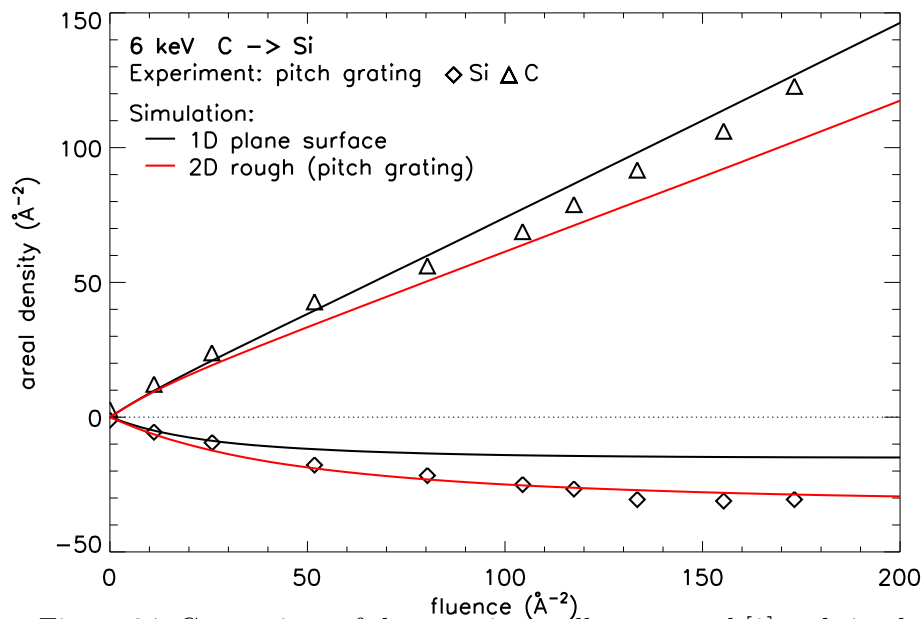


Figure 24: Comparison of the experimentally measured [9] and simulated fluence dependent variation of Si and C areal density normal angle of incidence

8. Two-dimensional examples with SDTrimSP-2D

The fluence dependent deposition of carbon is shown in Fig. 25. The simulation shows the penetration-depth of carbon. At the end of the calculation the surface is completely covered with carbon.

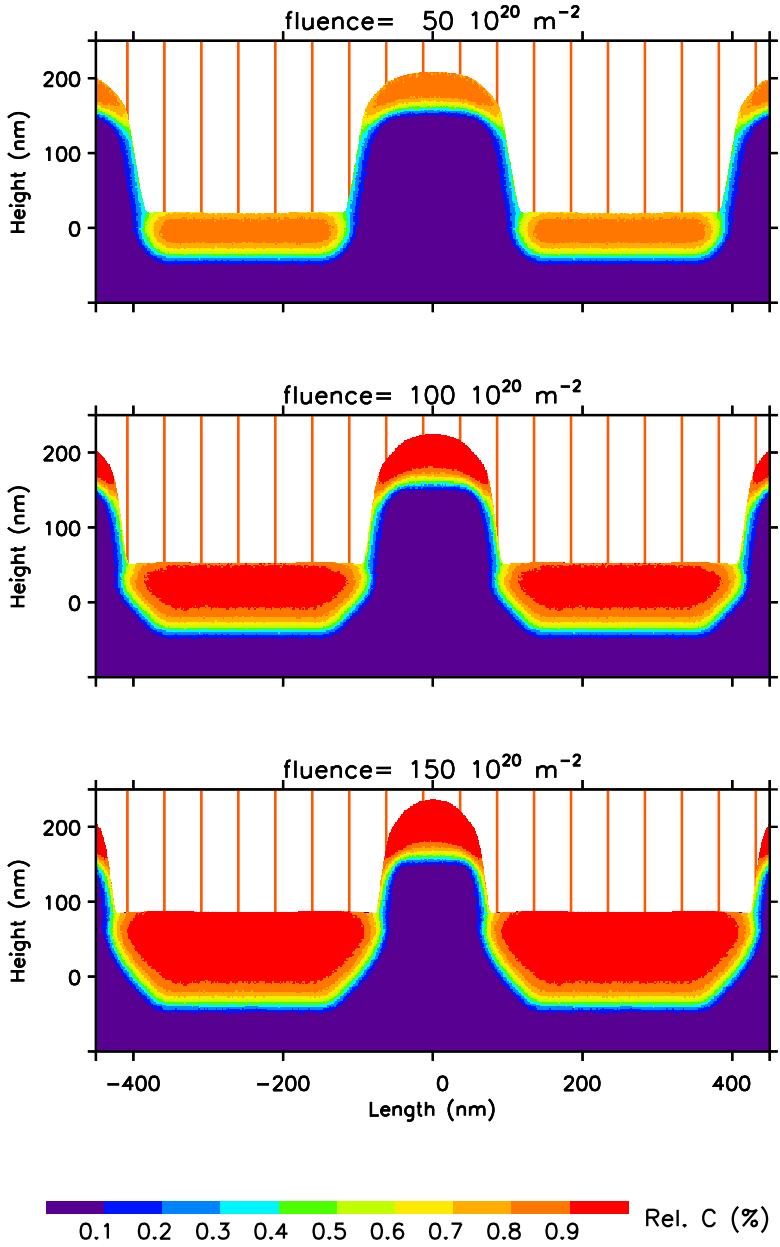


Figure 25: Simulated fluence dependent deposition of Carbon at 6 keV C on Si target normal angle of incidence

### 8.4. Influence of resolution

If the grid resolution of a target with inclined surface is insufficient the value of the sputter yield is more and more close to the value of a non-inclined surface, Fig. 27. The inclination of the surface is  $\gamma$ . The size of the grid should be much smaller than the extension of the trajectories, see Fig. 26.

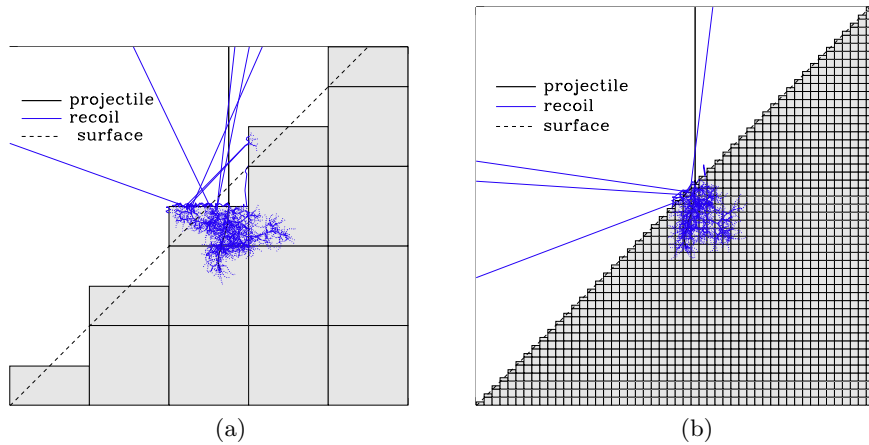


Figure 26: Trajectories at inclined plane of  $\gamma = 45^\circ$  (dash line) with a grid resolution (a)  $dx=dz=100 \text{ \AA}$  and (b)  $dx=dz=10 \text{ \AA}$ .

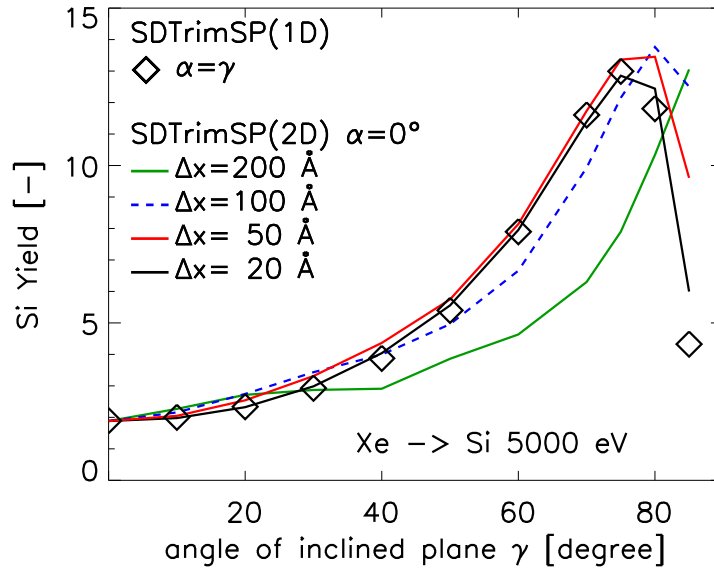


Figure 27: Sputter Yield of Si dependent on angle of incidence  $\alpha$  at different grid resolutions

8. Two-dimensional examples with SDTrimSP-2D

8.5. 5000 eV W on C-target with incidence angle of  $\alpha=30^\circ$

Another interesting case is the bombardment with an incident angle greater than zero, like the bombardment of C by W with an incident angle of  $\alpha = 30^\circ$ . The fluence dependent deposition of tungsten and the appearance of a hole is shown in Fig. 28.

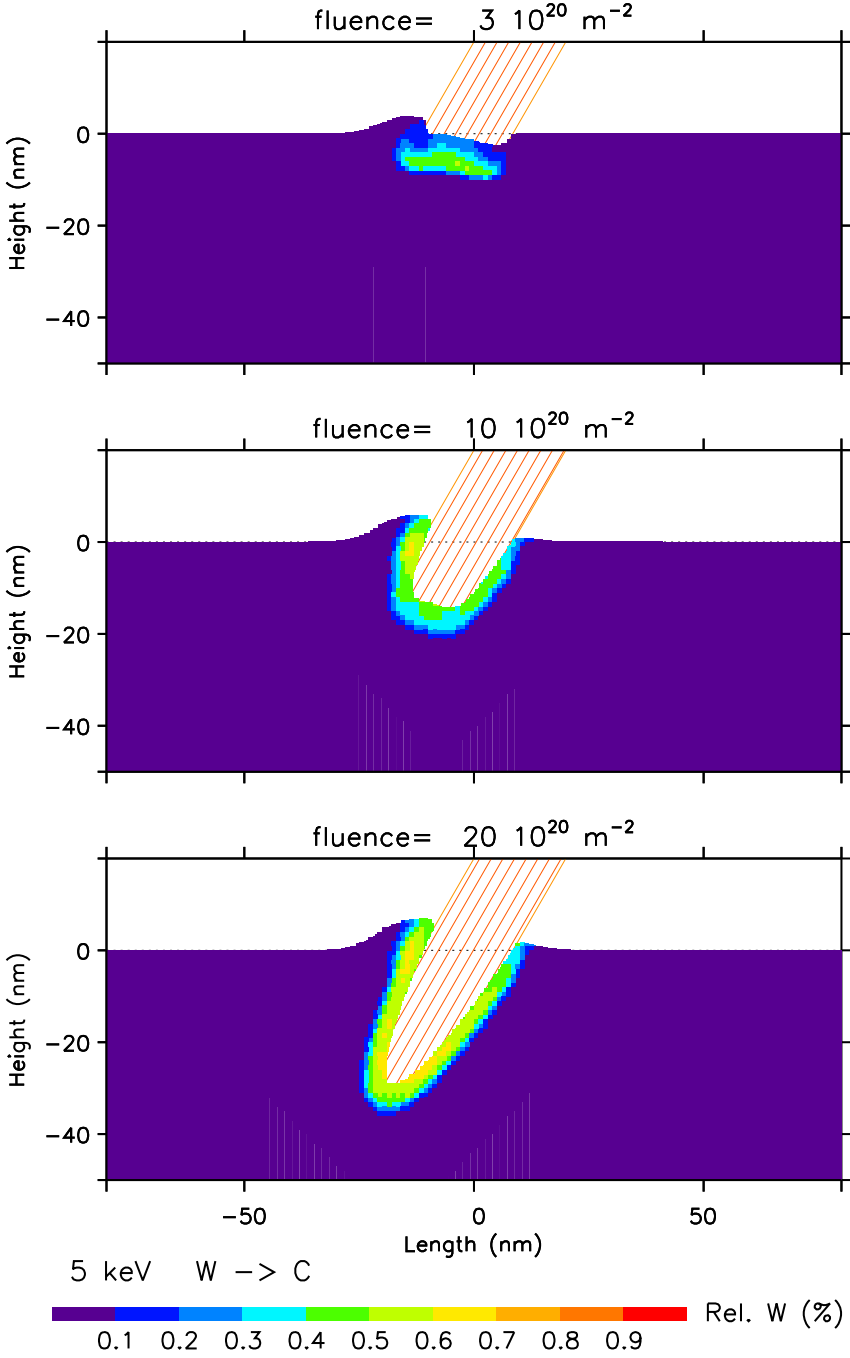


Figure 28: Simulated fluence dependent deposition of tungsten at 5 keV W on C target for an incident angle of  $\alpha = 30^\circ$

## References

- [1] W. Eckstein, *Computer Simulation of Ion-Solid Interactions*, Springer Series in Material Science, Vol. 10, Springer Berlin, Heidelberg 1991
- [2] W. Moeller, W. Eckstein, *TRIDYN-Binary Collision*, IPP 9/64, 1988
- [3] W. Moeller, W. Eckstein, *Tridyn-fzr-2001*, 2001
- [4] H. Schlager, W. Eckstein, *The scattering Integrals*, IPP 9/69, 1991
- [5] A. Mutzke, R. Schneider, W. Eckstein, R. Dohmen, SDTrimSP Version 5.0 IPP Report 12/8, Garching, (2011)
- [6] I. Bizyukov, A. Mutzke, R. Schneider, J. Davis, *Evolution of the 2D surface structure of a silicon pitch grating under argon ion bombardment: experiment and modeling*, Nucl. Instrum. Meth. vol.B 268, pp. 2631-2638, 2010
- [7] I. Bizyukov, A. Mutzke, R. Schneider, A.M. Gigler, K. Krieger, *Morphology and changes of elemental surface composition of tungsten bombarded with carbon ions*, Nucl. Instrum. Meth. vol.B 266, pp. 1979-1986, 2008
- [8] A. Mutzke, I. Bizyukov, R. Schneider, J. Davis, *Nano-scale modification of 2D surface structures exposed to 6 keV carbon ions: Experiment and modeling*, Nucl. Instrum. Meth. vol.B 269 pp. 582589,(2011)
- [9] A. Mutzke, I. Bizyukov, H. Langhuth, M. Mayer, K. Krieger, R. Schneider, *Study of interaction of C+ ion beam with a Si pitch grating on a macro-scale level*, Nucl. Instrum. Meth. vol.B 293 pp.1115 (2012)

A. Units of physics terms in the code

**A. Units of physics terms in the code**

name	symbol	unit
mean free path length	$l_m$	$\text{\AA} = 10^{-10}m$
random number	$R_{random}$	–
impact parameter	$P_{impact}$	$\text{\AA} = 10^{-10}m$
number of atoms	$N$	–
number of projectiles	$n_r$	–
fluence	$flc$	$atom/\text{\AA}^2$
geometry of cell	$\Delta x, \Delta y, \Delta z$	$\text{\AA}$
pseudo time	$\tau$	$s$
beam area	$A_{flc}$	$\text{\AA}^2$
volume	$V$	$\text{\AA}^3 = 10^{-30}m^3$
volume-change	$\Delta V$	$\text{\AA}^3$
atomic fraction	$qu$	–
length	$L$	$\text{\AA} = 10^{-10}m$
density	$\varrho$	$g/\text{\AA}^3 = g/10^{-30}m^3$
atomic density	$\varrho_0$	$atoms/\text{\AA}^3$
energy	$E$	$eV$

Table 2: Units of physics terms in the code

## B. Search algorithm of local cell

The mapping of particles to their corresponding cells is essential, hence the algorithms used in SDTrimSP and SDTrimSP-2D are described in the following section.

### SDTrimSP

The search algorithm of the actual layer in SDTrimSP is very simple and sketched in Fig. 29. The x-coordinate ( $x_p$ ) of every particle is known, so that a simple comparison is possible.

The particle is outside the target if  $x_p$  is smaller than  $-d_{refraction}$ . It is part of the refraction layer if  $x_p < 0$  and  $x_p > -d_{refraction}$  is true. If  $x_p > 0$  and  $x_p < -d_1$  than the particle is in the first layer and so on.

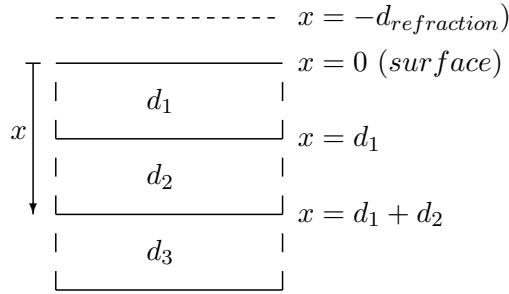


Figure 29: Depth geometry of 1D-target with three layers ( $d_1, d_2, d_3$ )

### SDTrimSP-2D

The search algorithm of the actual cell in SDTrimSP-2D, in which the particle exists or moves, is computed in the subroutine `'xyz_in_cel'`. A particle can be inside a cell (black line) but also inside of the refraction-thick (red dash line), due to a planar surface potential, Fig. 30. In all other cases it is outside of the target. The x- and z-coordinates of particle ( $x_p, z_p$ ) and the coordinates of each mesh ( $x_i, z_k$ ) are known.

The search algorithm can be divided into different steps or cases, that are shown (Fig. 30) and in the following:

In most cases, it is known in which cell the particle was located before it moves ( $N_{r-cell} \neq 0$ ). Hence, case 1 is the trivial test if the particle is still in the initial cell.

If the cell number is not known or the test of case 1 is negative, the mesh-number is determined due to particles coordinates in x and z direction ( $x_{i-1} < x_p < x_i, z_{k-1} < z_p < z_k$ ).



B. Search algorithm of local cell

Exist a cell in this mesh (index-cell  $\neq 0$ ) it is checked whether the particle is in this cell (case 2) or inside the refraction-zone (case 3).

If this test is also negative (no cell inside the mesh, index-cell = 0) then all cells are checked in the neighboring area. The cases (4-11) are top, bottom, left, right, top-left, top-right, bottom-left and bottom-right. It is possible that the particle is inside of the refraction-zone of other cells.

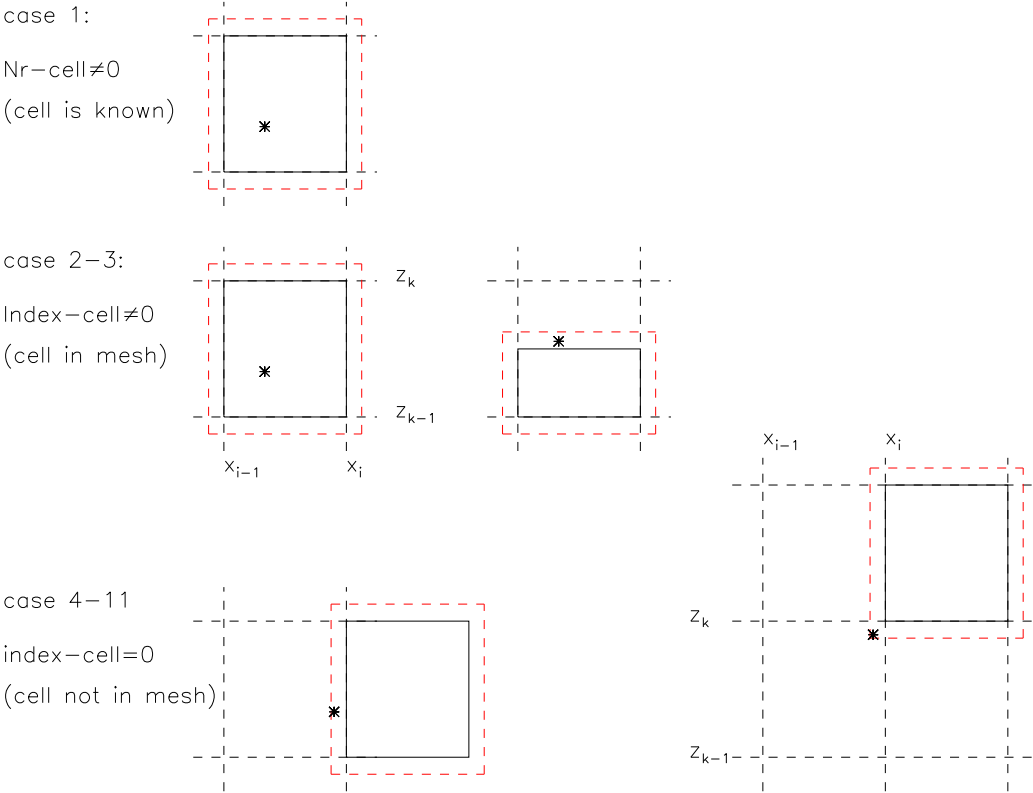


Figure 30: Cases of location of a particle in program. Red dash line is the refraction-zone at surface, due to a planar surface potential

### C. Geometry of a binary collision

Fig. 31 shows the points  $P_0$ ,  $P_1$  and  $P_3$  and  $T_1$  and  $T_2$  where computing takes place in the programs 'projectile' and 'recoil'.

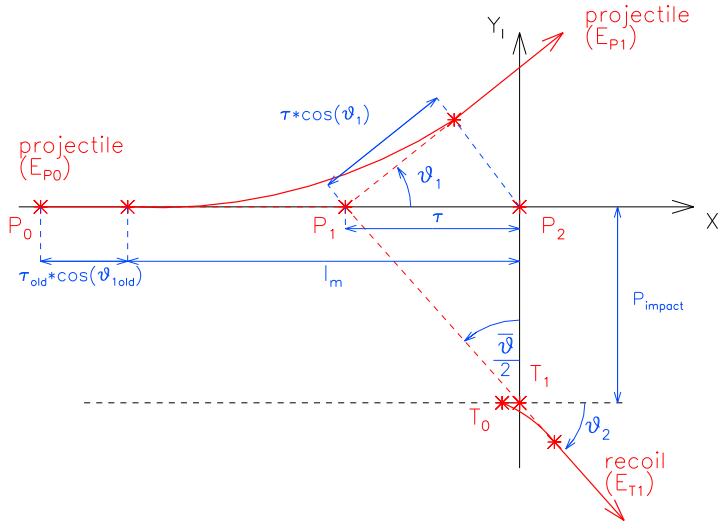


Figure 31: Trajectories of the projectiles and recoil particle in the laboratory system

## C. Geometry of a binary collision

### C.1. Gaussian distribution of incidence angle

Input values are case\_alpha=5, alpha0 and dalpha0. The polar angle alpha0 ( $\alpha_0$ ) has a range of  $[0 \dots 90^\circ]$ . The azimuthal angle  $\varphi_{azi} = 0^\circ/180^\circ$ .

$$\begin{aligned} \text{mean: } \alpha_0 &= \text{alpha0} = \frac{1}{n} \cdot \sum \alpha \\ \text{standard deviation: } \Delta\alpha &= \text{dalpha0} = \frac{1}{n-1} \cdot \sum (\alpha - \alpha_0) \\ \text{frequency function: } p &= \frac{1}{\sqrt{2 \cdot \Pi \Delta\alpha^2}} \cdot e^{-\frac{(\alpha - \alpha_0)^2}{2\Delta\alpha^2}} \end{aligned}$$

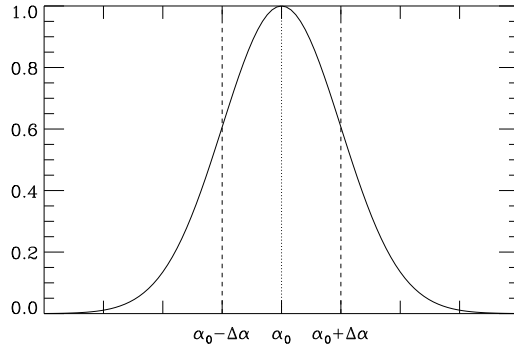


Figure 32: Gauss frequency distribution ( $p \cdot \sqrt{2 \cdot \Pi \Delta\alpha^2}$ ) of polar angles

### C.2. Gaussian distribution of incidence energy

Input values are case\_e0=5, e0\_beam and de0\_beam. The beam energy e0\_beam ( $e_0$ ) has a range of  $[1eV \dots 100MeV]$ .

$$\begin{aligned} \text{mean: } e_0 &= \text{e0\_beam} = \frac{1}{n} \cdot \sum e \\ \text{standard deviation: } \Delta e &= \text{de0\_beam} = \frac{1}{n-1} \cdot \sum (e - e_0) \\ \text{frequency function: } p &= \frac{1}{\sqrt{2 \cdot \Pi \Delta e^2}} \cdot e^{-\frac{(e - e_0)^2}{2\Delta e^2}} \end{aligned}$$

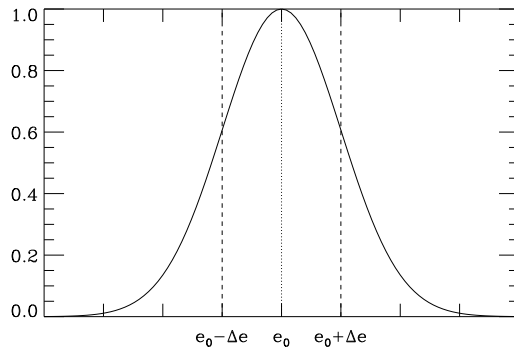


Figure 33: Gauss frequency distribution ( $p \cdot \sqrt{2 \cdot \Pi \Delta e^2}$ ) of energy (case 6)

## D. Global parameters

parameter	value	description	program
ncpm	8	max. number of species	param.F90
pemax	256	max. number of cores for parallelization	work.F90
ntqmax	1000000	memory request limit for coll. cascades	default.F90

Table 3: Global parameters (Values are set in the program)

## E. Input variables in 'tri.inp'

### E.1. Necessary input variables in 'tri.inp'

The sequence of input values in the input file is arbitrary (namelist), Tab. 4 -11

variable	description
alpha0(ncp)	angle of incidence (degree) of ncp species in case_alpha=0,5
e0(ncp)	energies (eV) of projectiles (qubeam > 0.) for case_e0=0,1,4 $kT = ttemp \cdot boltzm$ [eV] (e0_beam < 0) of projectiles for case_e0=2,3 $kT = e0\_beam$ [eV] (e0_beam(> 0) of projectiles for case_e0=2,3
flc	incident fluence ( $10^{16}$ atoms/cm <sup>2</sup> or atoms/Å <sup>2</sup> )
case_geo	flag of geometry and input surface = 10 : one dimensional = 20 : two dimensional, periodic, without stretch = 21 : two dimensional, periodic, without profile (smooth surface) = 23 : two dimensional, periodic, with surface profile in rauh.inp, left: x=0 = 24 : two dimensional, periodic, with surface profile in rauh.inp left: x=0, left values =right values is set in program = 25 : two dimensional, periodic, with surface profile in rauh.inp, centre: x=0 = 27 : two dimensional, periodic, with surface profile sin(x), centre: x=0

Table 4: Necessary input variables (no default values)

E. Input variables in 'tri.inp'

variable	description
ipot	interaction potential: = 1 : KrC = 2 : Moliere = 3 : ZBL = 4 : Na-Ya = 5 : Si-Si = 6 : power
isbv	surface binding model, determines the composition dependent surface binding energy sbv(ncp,ncp) from the elemental surface binding energies e_surfb(ncp) taken from table1 = 1 : sbv(ip,jp)=e_surfb(jp) for ip=jp = 2 : sbv(ip,jp)=e_surfb(jp) for all ip, jp = 3 : sbv(ip,jp)=0., if e_surfb(ip)=0 or e_surfb(jp)=0 sbv(ip,jp)=0.5*(e_surfb(ip)+e_surfb(jp)) = 4 : sbv(ip,jp)=f(e_surfb,qu,delta hf) for solid/solid compound = 5 : sbv(ip,jp)=f(e_surfb,qu,delta hf,delta hd) solid/gas compound
ncp	number of species (projectiles + target species)
maxhist	number of histories (projectiles)
qu.beam(ncp)	projectile atomic fractions (in incident beam) of ncp species, qu.beam > 0. , Note: sum(qubeam(1:ncp))=1 qu.beam ≤ 1. for projectiles, qu.beam = 0. for target atoms
qu.tar(ncp)	initial target atomic fractions of ncp species in case of homogenous initial composition (iq0 = 0)
symbol(ncp)	ncp chemical symbols of elements according to table1 (special symbol: 'H','D','T','He3','He','C_g','C_f','C_d')
two.comp	symbol of two-component target according to table.compound (e.g. two_comp ='Ta2O5') Note: only selected compounds in table.compound

Table 5: Necessary input variables (no default values)

**E.2. Optional input variables in 'tri.inp'**

These values have default values (see default\_init.txt). If values different from the default values are needed, then these values have to be given explicitly in the input file.

variable	default value	description
angleinp	'./'	directory of inputfile 'angle.inp' (see also: layerinp, tableinp, energyinp)
case_alpha	0	flag for the choice of the angle of incidence = 0 : angle of incidence (degree) counted from the surface normal (polar angle) is fixed $alpha0 > 0$ : $\alpha_{pol} = +alpha0$ , $\varphi_{azi} = 0^\circ$ $alpha0 < 0$ : $\alpha_{pol} = -alpha0$ , $\varphi_{azi} = 180^\circ$ = 1 : random distribution of angles $\alpha_{pol} = 0...alpha0$ , $\varphi_{azi} = 0...360^\circ$ = 2 : cosine distribution of angles of incidence $\alpha_{pol} = 0...90^\circ$ , max: by $45^\circ$ , $\varphi_{azi} = 0...360^\circ$ = 3 : cosine distribution of angles of incidence $\alpha_{pol} = 0...90^\circ$ , max: by $0^\circ$ , $\varphi_{azi} = 0...360^\circ$ = 4 : input of a given incident angular distribution from file angle.inp ( $\varphi_{azi} = 0^\circ$ ) = 5 : polar angle ( $\alpha_{pol}$ ) has a gaussian distribution $\varphi_{azi} = 0^\circ/180^\circ$ input: $\alpha_0 = alpha0$ and $\Delta\alpha = dalpha0$
case_e0	0	flag for the choice of the incident energy = 0 : fixed incident energies(eV) of projectiles (qubeam>0) = 1 : input of a given energy distribution from file energy.inp = 2 : temperature (eV) of a Maxwellian velocity distribution $kT = ttemp \cdot boltzm$ [eV] ( $e0\_beam < 0$ ) of projectiles $kT = e0\_beam$ [eV] ( $e0\_beam (> 0)$ ) of projectiles = 3 : temperature (eV) of a Maxwellian energy distribution $kT = ttemp \cdot boltzm$ [eV] ( $e0\_beam < 0$ ) of projectiles $kT = e0\_beam$ [eV] ( $e0\_beam (> 0)$ ) of projectiles = 5 : energy has a gaussian distribution input: $e_0 = e0\_beam$ and $\Delta e = de0\_beam$

Table 6: Optional input variables with default values

E. Input variables in 'tri.inp'

variable	default value	description
case_start	3	input of start position of projectiles ( x_start, y_start z_start) = 1 : x_start, y_start and z_start are given in tri.inp $z(\text{start}) = \text{cel}(x_{\text{start}}, y_{\text{start}}, z_{\text{start}}) + \text{thick\_deflec}$ = 2 : x_start, y_start and z0_rel are given in tri.inp $z(\text{start}) = z(\text{surface}) + z0_{\text{re}}$ = 3 : x_start, y_start and z_start are constant and given in tri.inp = 4 : y_start and z_start are constant and given in tri.inp x_start and dx_start are in tri.inp $x = x_{\text{start}} - 0.5 * dx_{\text{start}} + \text{random} * dx_{\text{start}}$ = 5 : program intern x-distribution
charge(ncp)	0	charge of species if case_e0=2,3 and sheath>0 (plasma) $\geq 1$ . for qubeam>0 (projectiles) = 0. for qubeam=0 (target atoms)
dalpha0(ncp)	1	$\Delta\alpha$ [degree] for gaussian distribution, see case_alpha =5
de0_beam(ncp)	0	$\Delta e$ [eV] for gauss distribution, see case_e0=5
deltahd(ncp)		heat of dissociation (eV) of a molecular target default from table1
deltahf	table	heat of formation (eV) of a molecular target default from table1
diff_koeff1(ncp)	1.0	transportcoefficient if loutgas true [ $\text{\AA}^3/\text{ion}$ ] (see also: loutgas)
diff_koeff2(ncp)	1.0	diffusionskoefficient if loutgas true [ $\text{\AA}^4/\text{ion}$ ] (see also: loutgas)
dns_0(ncp)	table	atomic density ( $\text{atoms}/\text{\AA}^3$ ) of ncp elements; default from table1
dx_start	0.	X-Width of the beam [ $\text{\AA}$ ]
e_bulkb(ncp)	0.	bulk binding energy; if e_bulkb>0., e_bulk has to be subtracted from the surface binding energy e_surfb
e_cutoff(ncp)	table	cutoff energy (eV) of ncp species; defaults from table1 (0.05 eV for noble gases; 1 eV for H, D, T; e_surf - 0.05 eV for selfbombardment)
e_displ(ncp)	table	displacement energy (eV); default from table1 (if in table1 e_displ=0 then e_displ=15)
e_surfb(ncp)	table	surface binding energy (eV) (heat of sublimation);
energyinp	'./'	directory of inputfile 'energy.inp' (see also: layerinp, tableinp, angleinp)
flc_flux	1.0	flux [ $\text{atoms}/(\text{\AA}^2 \text{ s})$ ], only to calculate time or fluence
flc_time	-1	time [s], only to calculate flux or fluence

Table 7: Optional input variables with default values (continue)

variable	default value	description
geo_x(2)	500	maximal width of target to right without(1) and with(2) cell-expansion, unit = [Å]
geo_z(2)	500	range of target=[-geo_x(2)..-geo_x(1)..geo_x(1)..geo_x(2)] maximal depth of target without (1) and with (2) cell-expansion, unit=[Å] range of depht=[0..geo_z(1)..geo_z(2)]
geo_dx	10	cell-width in X, [Å]
geo_dz	10	cell-width in Z, [Å]
geo_tau	1.0	geo_tau= $\Delta\tau$ relaxations coefficient, [s/Å <sup>2</sup> ]
ihist_out	-1	control output, determines the outputfiles: E0_31_target.dat, E0_34_moments.dat, partic*.dat, trajec*.dat and restart_file = -1 : output after each fluence step of maxhist/100, 100 fluence steps = 0 : output only after the last fluence step > 0 : output after each ihist_out'th fluence step and last step
iintegral	0	integration method = 0 : MAGIC, only valid for KrC, ZBL, Moliere not recommended = 1 : Gauss-Mehler quadrature, ipivot $\geq 8$ = 2 : Gauss-Legendre quadrature, ipivot $\leq 16$ recommended
inel0(ncp)	3	inelastic loss model = 1 : Lindhard-Scharff; nessary condition: $E < 25 \cdot Z^{4/3} \cdot M$ (in keV) where E, Z, M are the energy, the atomic number and the atomic mass of the moving particle = 2 : Oen-Robinson; nessary condition: $E < 25 \cdot Z^{4/3} \cdot M$ (in keV) = 3 : equipartition of 1 and 2 = 4 : high energy hydrogen (H,D,T) (energy > 25 keV) values from table3 = 5 : high energy helium (He3,He) (energy > 100 keV) values from table4

Table 8: Optional input variables with default values (continue)



E. Input variables in 'tri.inp'

variable	default value	description
ioutput_hist(6)	10	number of traced trajectories for: stopped, backscattered and transmitted projectiles, stopped, backscattered, transmission sputtered recoils (see also: ltraj_p, ltraj_r)
ioutput_part(6)	10	number of traced particles for: stopped, backscattered and transmitted projectiles, stopped, backscattered, transmission sputtered recoils (see also: lparticle_p, lparticle_r)
ipivot	16	number of pivots in the Gauss-Mehler and Gauss-Legendre integration, the minimum number is 4 (larger numbers increase the computing time)
iq0	0	initial composition flag < 0 : initial depth dependent composition taken from file layer.inp = 0 : initial composition homogeneous, one layer with constant depth intervals
irand	1	random seed
irc0	-1	flag for subthreshold recoil atoms < 0 : subthreshold recoil atoms free ≥ 0 : subthreshold atoms bound
isot(ncp)	0	flag for isotope mass = 0 : natural isotope mixture (mass from table1) = 1 : isotope masses and natural abundances from table2 (valid for projectiles as well as for target species)
i_two_comp	1	method to determine the densities dns0(:) from the compound density in a two-component target (table.compound) =1 : dns0 for the first target species is set equal to the elemental density; necessary if the second element is a gas (e.g. Ta2O5) =2 : dns0 for the second target species is set equal to the elemental density =3 : iterative determination of both dns0(:); recommended if the elemental densities are different
iwc=2	2	number of ring cylinders for weak simultaneous collisions for projectiles; for high energies (MeV H or He) iwc can be reduced to 1 or 0 to reduce computing time
iwcr=2	2	number of ring cylinders for weak simultaneous collisions for recoils

Table 9: Optional input variables with default values (continue)

variable	default value	description
layerinp	'./'	directory of inputfile 'layer.inp' (see also: tableinp, angleinp, energyinp)
lchem_ch	.false.	calculation with chemical erosion H on C, D on C
lmatrices	.false.	.true. : output of matrices, if idrel $\neq$ 0 .false. : no matrix output
loutgas	.false.	calculation with outgasing transport and diffusion (see also diff_koeff1,diff_koeff2)
lparticle_p	.false.	.true. : output of projectile information .false. : no output of projectile information (see also: ioutput_part)
lparticle_r	.false.	.true. : output of recoil information .false. : no output of recoil information (see also: ioutput_part)
ltraj_p	.false.	.true. : output of projectile trajectories .false. : no output of projectile trajectories (see also: numb_hist, ioutput_hist)
ltraj_r	.false.	.true. : output of recoil trajectories .false. : no output of recoil trajectories (see also: numb_hist, ioutput_hist)
l_write_restart	.false.	.true. : output of restart-file
nm_mol	-1	=-1 : not a molecular target > 1 : number of atoms in a two-component molecule
nr_pproj	10	number of projectiles between two target updates (idrel = 0)
numb_hist	20	number of traced trajectories of projectiles and recoils
qumax(ncp)	1.	maximum atomic fractions in the target for ncp species, if idrel=0
rhom_mol		atomic density of a two-component target; default from table.compound [ $g/cm^3$ ]
sfin	0.	= 0 : no inelastic energy loss outside the target surface ( $x = 0.$ ) = 1 : inelastic energy loss outside the target surface ( $-su > x > 0.$ )
shth	0.	= 0 : no sheath potential > 0 : sheath potential (eV), usually = $3 \cdot  e0  \cdot \text{charge}$ , only if case_e0=2,3 (Maxwellian distribution, plasma)
tableinp	'.//tables'	directory of inputfile for tables (see also: layerinp, angleinp, energyinp)

Table 10: Optional input variables with default values (continue)

E. Input variables in 'tri.inp'

variable	default value	description
tar_dynamic	true	mode of simulation = true : full dynamic calculation (like TRIDYN) = false : suppression of dynamic relaxation (like TRIM), full static calculation
ttemp	0.	target temperature, only of interest at high temperatures, it reduces the surface binding energy according to a Maxwellian energy distribution
tttt		text in tri.inp
x_start	0.	x starting position of projectile
z_start	0.	z starting position of projectile
z0_rel	0.	z starting position of projectile

Table 11: Optional input variables with default values (continue)

## F. Example of Inputfile 'tri.inp' and 'layer.inp'

Inputfile 'tri.inp' of 1D example He - > Ta<sub>2</sub>O<sub>5</sub>

```
1000 eV He - > Ta2O5
& TRI.INP
/
tttt='—elemements—'
      ncp = 3
      symbol = "He", "Ta", "O"
tttt='—target—'
      nm_mol = 7
      two_comp = "Ta2O5"
      qu_tar = 0.000, 0.285714, 0.714286
      qu_max = 0.000, 1.000, 0.714286
      inel0 = 5, 3, 3
      e_cutoff = 1.0 ,1.0 ,1.0
tttt='—control—'      maxhist = 100000
      ihist_out = 2000
      flc = 10
      nr_pproj = 10
      tar_dynamic = .true.
      ipot = 1
      isbv = 5
      iintegral = 2
      ipivot=8
tttt='—beam—'
      qu_beam = 1.000, 0.000, 0.000
      case_e0 = 0
      e0_beam = 1000, 0.00, 0.00
      case_alpha= 0
      alpha0 = 0.000, 0.000, 0.000
      case_start= 3
      x_start = 0.
      dx_start= 0.
      z_start = 100.
tttt='geometry: case_geo = 10 ... 1-D'
      case_geo = 10
      geo_x = 500. 500.
      geo_z = -500. -500.
      geo_dx = 1000.0
      geo_dz = 5.0
/
```

*F. Example of Inputfile 'tri.inp' and 'layer.inp'*

**Inputfile 'tri.inp' of 1D example Xe - > Si**

```
2000 eV Xe - > Si
& TRLINP
tttt='—elements—'
    ncp = 2
    symbol ="Xe", "Si"
tttt='—beam—'
    qu_beam = 1.000, 0.000
    case_e0 = 0
    e0_beam = 2000, 0.00
    case_alpha= 0
    alpha0 = 0.000, 0.000
tttt='—target—'
    qu_tar = 0.000, 1.000
    qu_max = 1.000, 1.000
    inel0 = 3, 3
    e_cutoff = 1.0 ,1.0
tttt='—control—'
    maxhist = 10000
    ihist_out = 100
    flc = 1
    nr_pproj = 40
    tar_dynamic = .true.
    ipot = 1
    isbv = 1
tttt='—beam—'
    case_start= 3
    x_start = 00.
    dx_start= 0.
    z_start = 1000.
tttt='geometry 10 one dim'
    case_geo = 10
    geo_x = 500. 500.
    geo_z = -500. -500.
    geo_dx = 1000.0
    geo_dz = 5.0
    loutgas = .true.
tttt=' diff_koeff'
    diff_koeff1 =1.7e06, 0.0
    diff_koeff2 = 200, 0.0
/
```

**Inputfile 'tri.inp' of 2D example Xe -> Si, see Fig. 19**

```
5000 eV Xe -> Si
& TRI_INP
tttt='—elements—'
      ncp = 2
      symbol = "Xe", "Si"
tttt='beam'
      qu_beam = 1.000, 0.000
      case_e0 = 0
      e0_beam = 5000, 0.00
      case_alpha= 0
      alpha0 = 0.000, 0.000
tttt=' control'
      flc = 2
      maxhist = 1000
      ihist_out = 500
      nr_pproj = 400
      ipot = 1
      isbv = 1
tttt='target'
      tar_dynamic = .true.
      qu_tar = 0.000, 1.000
      inel0 = 3 , 3
      irc0=1
      e_cutoff= 1.0 , 1.0
tttt='beam geometry 4... given x:x+dx*random, z_start=constant'
      case_start= 4
      x_start = 5.
      dx_start = 50.
      z_start = 500.
tttt='target 21..without profile '
      case_geo = 21
      geo_x = 500. 500.
      geo_z = -500. -500.
      geo_dx = 10.0
      geo_dz = 10.0
      loutgas = .true.
      diff_koeff1 =1.7e06, 0.0
      diff_koeff2 = 200, 0.0
/
```

F. Example of Inputfile 'tri.inp' and 'layer.inp'

**Inputfile 'tri.inp' of 2D example Ar - > Si**, see Fig. 20 - 23

```
6 keV Ar - > Si
& TRILINP
tttt='—element—'
    ncp = 3
    symbol = "Ar", "Si", "Ta"
tttt=' beam'
    qu_beam = 1.00, 0.00 , 0.00
    case_e0 = 0
    e0_beam = 6000.00, 0.00, 0.00
    case_alpha= 0
    alpha0 = 42.000, 0.000, 0.000
tttt='control'
    maxhist = 24000
    ihist_out = 1000
    flc = 60
    nr_pproj = 1024
    ipot = 1
    isbv = 1
    irc0=1
tttt='target'
    tar_dynamic = .true.
    inel0 = 3, 3, 3
    qu_tar = 0.00, 1.00, 0.00
    e_cutoff = 2.0 ,2.0 ,2.0
tttt=' 0... homogen 1... initial_composition.inp'
    iq0=1
tttt=' outgasing'      loutgas = .true.
    diff_koeff1 = 1.00e5, 0
    diff_koeff2 = 50 , 0
tttt=' 4... given x:x+dx*random, z_start=constant'
    case_start= 4
    x_start = 0.
    dx_start= 5000.
    z_start = 2500.
tttt='25 with profile rauh.inp'
    case_geo=25
    geo_x = 2500., 2500.
    geo_z = -5000., -5000.
    geo_dx = 20.0
    geo_dz = 20.0
/
```

**Inputfile 'rauh.inp' of 2D example Ar - > Si, see Fig. 20 - 23**

pitch grating (x,y Values)

```
6
-2600.0 0000.0
-1350.0 0000.0
-1100.0 2000.0
1100.0 2000.0
1350.0 0000.0
2600.0 0000.0
```